A Family of Finite Volume Eulerian-Lagrangian Methods for Two-dimensional Conservation Laws

Fayssal Benkhaldoun[*] Saida Sari[†] Mohammed Seaid[‡]

Abstract

We develop a family of finite volume Eulerian-Lagrangian methods for the solution of nonlinear conservation laws in two space dimensions. The proposed approach belongs to the class of fractional-step procedures where the numerical fluxes are reconstructed using the modified method of characteristics, while an Eulerian method is used to discretize the conservation equation in a finite volume framework. The method is simple, accurate, conservative and it combines advantages of the modified method of characteristics to accurately solve the nonlinear conservation laws with an Eulerian finite volume method to discretize the equations. The proposed finite volume Eulerian-Lagrangian methods are conservative, non-oscillatory and suitable for hyperbolic or non-hyperbolic systems for which Riemann problems are difficult to solve or do not exist. Numerical results are presented for an advection-diffusion equation with known analytical solution. The performance of the methods is also analyzed on several applications in Burgers and Buckley-Leverett problems. The aim of such a method compared to the conventional finite volume methods is to solve nonlinear conservation laws efficiently and with an appropriate level of accuracy.

Keywords. Finite volume method; Modified method of characteristics; Eulerian-Lagrangian schemes; Conservation laws; Convection-diffusion problems

1 Introduction

It is the goal of this paper to construct an efficient finite volume Eulerian-Lagrangian schemes for the two-dimensional equations of conservation laws

$$\begin{aligned} \frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} + \frac{\partial g(u)}{\partial y} &= 0, && (x,y) \in \mathbb{R}^2, \quad t > 0, \\ u(0,x,y) &= u_0(x,y), && (x,y) \in \mathbb{R}^2, \end{aligned} \tag{1}$$

where u is the unknown solution, u_0 is a given initial data, $f(u)$ and $g(u)$ are nonlinear flux functions assumed to be differentiable and may depend on space and time as well. The conservation laws (1) have been used in the literature to model a wide phenomena of practical applications in science and engineering, including gas dynamics, flows in porous media and transport processes, etc. The numerical modelling of these problems requires the design of accurate and efficient tools to capture the fine solution features. Solving numerically the nonlinear equations (1) is still a considerable task in the convection-dominated situations, see for example [8, 10]. In these cases, the convection is a source of computational difficulties and nonphysical oscillations. It is also well known that solutions of the equations (1) present steep fronts, sharp discontinuities, boundary layers and shocks, which need to be resolved accurately in applications and often

[*]LAGA, Université Paris 13, 99 Av J.B. Clement, 93430 Villetaneuse, France (fayssal@math.univ-paris13.fr)

[†]LAGA, Université Paris 13, 99 Av J.B. Clement, 93430 Villetaneuse, France (sari@math.univ-paris13.fr)

[‡]School of Engineering and Computing Sciences, University of Durham, South Road, Durham DH1 3LE, UK (m.seaid@durham.ac.uk)

cause severe numerical difficulties. Fully Eulerian finite volume methods use fixed grids and incorporate some upwinding discertization in their formulations to reconstruct the numerical fluxes. Among the class of Eulerian finite volume methods are the Monotone Upstream-centered Schemes for Conservation Laws (MUSCL) and also include many other methods such as the high-resolution methods from computational fluid dynamics, in particular, the Godunov methods and the essentially non-oscillatory (ENO) methods, see for instance [8, 13] among others. However, the reconstruction of numerical fluxes in these methods requires solution of Riemann problems at the control volumes that may become computationally demanding for conservation laws in two and three space dimensions. In addition, these methods can not be applied to conservation laws for which Riemann problems are not solvable or do not exist as those associated with the non-hyperbolic problems. Modified method of characteristics or semi-Lagrangian (SLAG) methods as known in the meteorological community, on the other hand, make use of the transport nature of the conservation laws. They combine the fixed Eulerian grids with a particle tracking along the characteristics of the governing equations. The Lagrangian treatment in these methods greatly reduces the time truncation errors in the Eulerian methods. In addition, these methods alleviate the restrictions on the Courant number, thus allowing for large time steps in the simulations. For a review on numerical methods for advection-dominated equations, we refer the reader to [6, 10, 18] and further references are therein. However, the conventional SLAG methods are known to be neither conservative nor Total Variation Diminishing (TVD). Indeed, if the solution of the governing equations is expected to have sharp gradients, the numerical solution obtained by the conventional SLAG methods either develops spurious oscillations or is affected by a large artificial viscosity. Spurious oscillations and artificial viscosity often deteriorate the accuracy of the solution, so the numerical solution may become physically unacceptable. Another severe limitation of standard SLAG methods is the failure to preserve the positivity of the numerical solutions. In the current study we present a class of finite volume Eulerian-Lagrangian methods which takes advantages from the conventional SLAG techniques in an Eulerian finite volume framework. The present methods offer several advantages over competing techniques for solving conservation laws. In contrast to traditional Eulerian finite volume methods, this technique incorporates the integration of the equations under study along the characteristics, such that the numerical fluxes are easily calculated. The present technique has a distinct advantage over the conventional SLAG methods as it is readily applicable to conservation laws and nonlinear convection problems. The method does not require any solver for Riemann problems and as it is an extension of the finite volume method, it is monotone, mass conservative, and it can exploit the large body of finite volume technology and software.

The object of this paper is to devise a numerical approach capable of accurately approximating solutions to linear and nonlinear equations of conservation laws in two space domains. The aim is to develop a simple and accurate family of finite volume Eulerian-Lagrangian methods that incorporate techniques from the modified method of characteristics into the reconstruction of numerical fluxes. Our main goal is to present a class of numerical methods that are simple, easy to implement, and accurately solves the nonlinear conservation laws without relying on Riemann problem solvers. Combining modified method of characteristics with finite volume method has been investigated for example in [11] for solving Newtonian contraction flows and in [16] for advection-diffusion problems. It should be stressed that, unlike the conservation laws (1), the governing equations in [11, 16] are casted in an advective form for which the conventional SLAG schemes are very well suited. A new combined modified finite volume method of characteristics has been recently proposed in [2] for the one-dimensional shallow water equations. This method has been extended in [1] for the numerical solution of morphodynamic problems in one space dimension and it is adapted in the current work for the numerical solution of the two-dimensional conservation laws (1). The proposed finite volume Eulerian-Lagrangian scheme belongs to the class of methods that employ only physical fluxes and averaged states in their formulations. It can be interpreted as a predictor-corrector scheme. In the corrector stage, the considered equations are integrated over an Eulerian time-space control volume whereas in the predictor stage, the conservation laws are rewritten in an advective form and integrated along the characteristics defined by the advection velocity field. The main features of such a finite volume Eulerian-Lagrangian scheme are on one hand, the capability to satisfy the conservation property resulting in numerical solutions free from spurious oscillations, and on the other hand, the achievement of strong stability and high accuracy for

numerical solutions containing shocks or discontinuities. These features are verified using several test examples of the two-dimensional conservation laws (1) including linear advection-diffusion equations, inviscid and viscous Burgers equations, and the Buckley-Leverett problem. Results presented in this paper show high resolution of the proposed finite volume Eulerian-Lagrangian schemes and permit the straightforward application of the method to more complex, physically based conservation laws. The present study represents a step towards the implementation of a modified method of characteristics for the finite volume solution of nonlinear conservation laws.

This paper is organized as follows. The family of finite volume Eulerian-Lagrangian methods is formulated in section 2. This section includes the Eulerian stage for the finite volume discretization and the Lagrangian stage for the reconstruction of the numerical fluxes. In Section 3, an analysis of stability is presented for the proposed method. Section 4 is devoted to the extension of the method for the numerical solution of nonlinear convection-diffusion problems. Numerical results are presented in Section 5. Our new approach is shown to enjoy the expected accuracy as well as the robustness. Section 6 contains concluding remarks.

2 Finite Volume Eulerian-Lagrangian Methods

In this section we formulate the finite volume Eulerian-Lagrangian method for the numerical solution of the conservation law (1). The method consists of two steps and can be interpreted as a predictor-corrector approach. The first step deals with the finite volume discretization of the equation while in the second step, reconstruction of the numerical fluxes is discussed.

2.1 The Eulerian stage

For the space discretization of the equations (1), we cover the spatial domain with rectangular cells $\mathcal{C}_{i,j} = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}]$ of uniform sizes Δx and Δy for simplicity in the presentation only. The cells, $\mathcal{C}_{i,j}$, are centered at $(x_i = i\Delta x, y_j = j\Delta y)$. We use the notations

$$U_{i\pm\frac{1}{2},j}(t) = u(t, x_{i\pm\frac{1}{2}}, y_j), \qquad U_{i,j\pm\frac{1}{2}}(t) = u(t, x_i, y_{j\pm\frac{1}{2}})$$

$$\text{and} \qquad U_{i,j}(t) = \frac{1}{\Delta x}\frac{1}{\Delta y}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}\int_{y_{j-\frac{1}{2}}}^{y_{i+\frac{1}{2}}} u(t,x,y)dxdy,$$

to denote the point-values and the approximate cell-average of u at the gridpoint $(t, x_{i\pm\frac{1}{2}}, y_j)$, $(t, x_i, y_{j\pm\frac{1}{2}})$, and (t, x_i, y_j), respectively. Integrating the equation (1) with respect to space over the control volume $\mathcal{C}_{i,j}$ shown in Figure 1, we obtain the following semi-discrete equation

$$\frac{\partial U_{i,j}}{\partial t} + \frac{F_{i+1/2,j} - F_{i-1/2,j}}{\Delta x} + \frac{G_{i,j+1/2} - G_{i,j-1/2}}{\Delta y} = 0, \tag{2}$$

where $F_{i\pm1/2,j} = f(U_{i\pm1/2,j})$ and $G_{i,j\pm1/2} = g(U_{i,j\pm1/2})$ are the numerical fluxes at the cell interfaces $x = x_{i\pm1/2}$ and $y = y_{i\pm1/2}$, respectively. The spatial discretization of equation (2) is complete when a time integration is performed and a numerical construction of the fluxes $F_{i\pm1/2,j}$ and $G_{i,j\pm1/2}$ is chosen. In general, this construction requires a solution of Riemann problems at the interfaces $x_{i\pm1/2}$ and $y_{i\pm1/2}$, see for example [8, 15] among others. From a computational viewpoint, this procedure is very demanding and may restrict the application of the method for which Riemann solutions are not available.

To integrate the equations (2) in time we divide the time interval into N subintervals $[t_n, t_{n+1}]$ with length $\Delta t = t_{n+1} - t_n$ for $n = 0, 1, \ldots, N$. We use the notation w^n to denote the value of a generic function w at

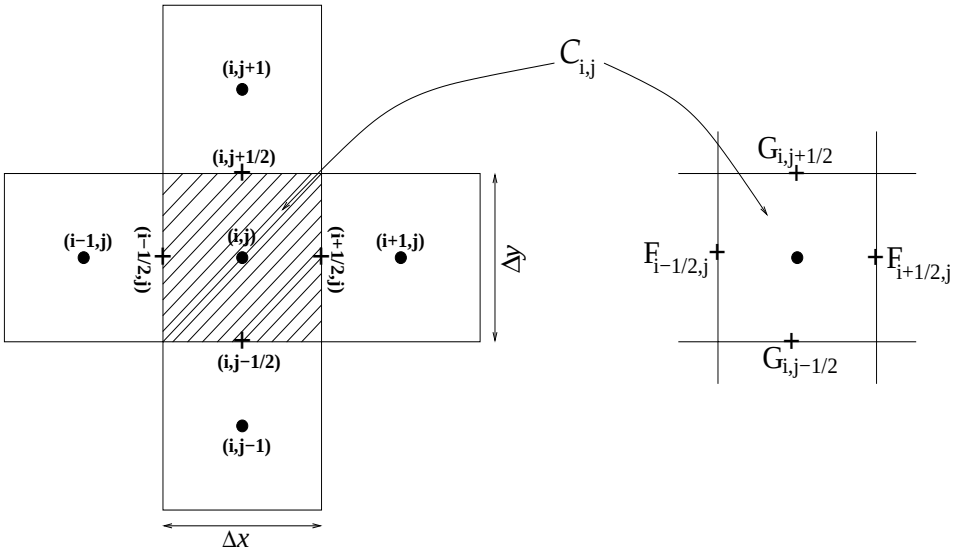

Figure 1: An illustration of control volumes $\mathcal{C}_{i,j}$ used in the proposed method.

time t_n. We may consider a first-order stepping integration scheme, in which the fully-discrete formulation of the conservation law (1) is given by

$$U_{i,j}^{n+1} = U_{i,j}^{n} - \frac{\Delta t}{\Delta x}\left(F_{i+1/2,j}^{n} - F_{i-1/2,j}^{n}\right) - \frac{\Delta t}{\Delta y}\left(G_{i,j+1/2}^{n} - G_{i,j-1/2}^{n}\right), \tag{3}$$

where $F_{i\pm1/2,j}^{n} = f(U_{i\pm1/2,j}^{n})$ and $G_{i,j\pm1/2}^{n} = g(U_{i,j\pm1/2}^{n})$. It should be stressed that because of the explicit treatment of the flux functions this scheme is conditionally stable. Implicit treatment of these treatment is also possible but this type of implicit schemes are rarely used for conservation laws as they involve solution of linear/nonlinear systems at each time step. Note that other high-order time stepping methods can also be applied in (2).

2.2 The Lagrangian stage

In the present study, we reconstruct the numerical fluxes $F_{i\pm1/2,j}$ and $G_{i,j\pm1/2}$ using the modified method of characteristics. The fundamental idea of this method is to impose a regular grid at the new time level and to backtrack the flow trajectories to the previous time level, see for example [5, 14, 21]. At the old time level, the quantities that are needed are evaluated by interpolation from their known values on a regular grid. Hence, the conservation law (1) can be rewritten in an advective form as

$$\frac{\partial u}{\partial t} + f'(u)\frac{\partial u}{\partial x} + g'(u)\frac{\partial u}{\partial y} = 0, \qquad (x,y) \in \mathbb{R}^2, \quad t > 0. \tag{4}$$

Thus, the characteristics associated with the equation (4) are solutions of the initial-value problems

$$\begin{aligned}
\frac{dX_{i+1/2}(\tau)}{d\tau} &= V\Big(\tau, X_{i+1/2}(\tau), Y_j(\tau)\Big), && \tau \in [t_n, t_n + \alpha\Delta t],\\
\frac{dY_j(\tau)}{d\tau} &= W\Big(\tau, X_{i+1/2}(\tau), Y_j(\tau)\Big), && \tau \in [t_n, t_n + \alpha\Delta t],\\
X_{i+1/2}(t_n + \alpha\Delta t) &= x_{i+1/2}, && Y_j(t_n + \alpha\Delta t) = y_j,
\end{aligned} \tag{5}$$

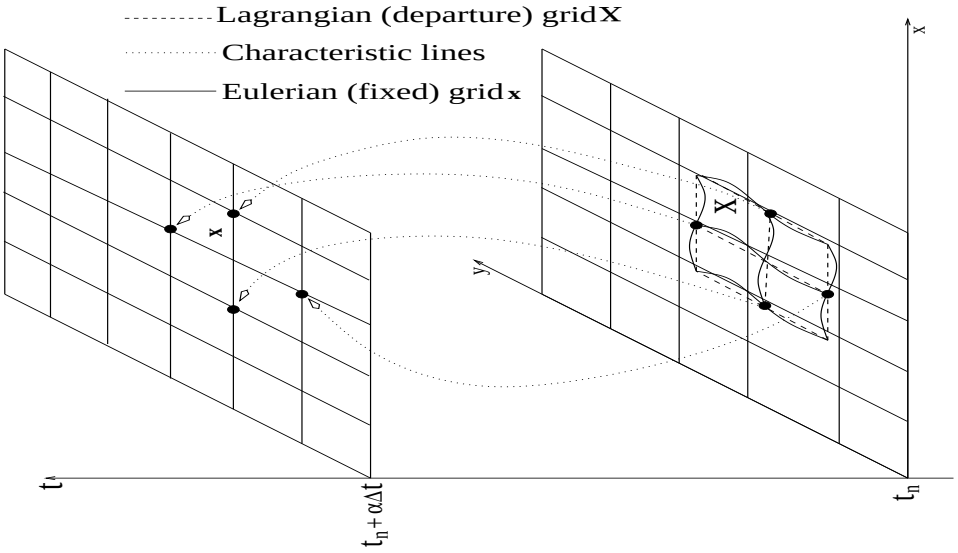

Figure 2: A schematic diagram showing the main quantities used in calculation of the departure points. The exact mapping is represented by a solid line and the approximate mapping with a dashed line. The doted lines represent the characteristics.

and

$$\begin{aligned}
\frac{dX_i(\tau)}{d\tau} &= V\Big(\tau, X_i(\tau), Y_{j+1/2}(\tau)\Big), \qquad \tau \in [t_n, t_n + \alpha\Delta t], \\
\frac{dY_{j+1/2}(\tau)}{d\tau} &= W\Big(\tau, X_i(\tau), Y_{j+1/2}(\tau)\Big), \qquad \tau \in [t_n, t_n + \alpha\Delta t], \qquad (6) \\
X_i(t_n + \alpha\Delta t) &= x_i, \qquad Y_{j+1/2}(t_n + \alpha\Delta t) = y_{j+1/2},
\end{aligned}$$

where $V = f'(U)$, $W = g'(U)$ and $\alpha \in]0, 1]$ is a parameter to be selected in the sequel. Note that $\mathbf{X}_{i+1/2,j}(\tau) = (X_{i+1/2}(\tau), Y_j(\tau))^T$ is the departure point at time τ of a particle that will arrive at the gridpoint $\mathbf{x}_{i+1/2,j} = (x_{i+1/2}, y_j)^T$ in time $t_n + \alpha\Delta t$ whereas, $\mathbf{Y}_{i,j+1/2}(\tau) = (X_i(\tau), Y_{j+1/2}(\tau))^T$ is the departure point at time τ of a particle that will arrive at the gridpoint $\mathbf{y}_{i,j+1/2} = (x_i, y_{j+1/2})^T$ in time $t_n + \alpha\Delta t$. The method of characteristics does not follow the flow particles forward in time, as the Lagrangian schemes do, instead it traces backwards the position at time t_n of particles that will reach the points of a fixed mesh at time $t_n + \alpha\Delta t$. By doing so, the method avoids the grid distortion difficulties that the conventional Lagrangian schemes have. In what follows we formulate the method for solving (5) while the solution of (6) can be carried out in a similar manner. Hence, the solution of (5) can be expressed in a vector form as

$$\mathbf{X}_{i+1/2,j}(t_n) = \mathbf{x}_{i+1/2,j} - \int_{t_n}^{t_n+\alpha\Delta t} \mathbf{V}\left(\tau, \mathbf{X}_{i+1/2,j}(\tau)\right) d\tau, \qquad (7)$$

where $\mathbf{V}_{i+1/2,j} = (V_{i+1/2,j}, W_{i,j})^T$ is the advective velocity. Accurate estimation of the characteristics $\mathbf{X}_{i+1/2,j}(t_n)$ is crucial to the overall accuracy of the Eulerian-Lagrangian methods. Some authors estimate $\mathbf{X}_{i+1/2,j}(t_n)$ by a second-order explicit Runge-Kutta scheme, which authors in [3, 20] have found is not accurate enough to maintain a particle on its curved trajectory. A better choice is a fourth order explicit Runge-Kutta scheme. However, it is inconvenient that the velocity has to be extrapolated at both, $t_{n+\frac{\alpha}{2}}$ and $t_{n+\alpha}$. In this paper, we used a method first proposed in [21] in the context of semi-Lagrangian schemes to integrate the weather prediction equations.

Let $\mathbf{d}_{i+1/2,j}$ denotes the displacement between a gridpoint on the new level, $\mathbf{x}_{i+1/2,j} = (x_{i+1/2}, y_j)^T$, and

the departure point of the trajectory to this point on the previous time level $\mathbf{X}_{i+1/2,j} = (X_{i+1/2,j}, Y_j)^T$,

$$\mathbf{d}_{i+1/2,j} := \mathbf{x}_{i+1/2,j} - \mathbf{X}_{i+1/2,j}(t_n),$$

if we use the mid-point rule to approximate the integral in (7), we have

$$\mathbf{d}_{i+1/2,j} = \alpha \Delta t \mathbf{V}\left(t_{n+\frac{\alpha}{2}}, \mathbf{X}_{i+1/2,j}(t_{n+\frac{\alpha}{2}})\right). \tag{8}$$

using

$$\mathbf{X}_{i+1/2,j}(t_{n+\frac{\alpha}{2}}) = \mathbf{x}_{i+1/2,j} - \frac{\alpha}{2}\mathbf{d}_{i+1/2,j},$$

and

$$\mathbf{V}\left(t_{n+\frac{\alpha}{2}}, \mathbf{X}_{i+1/2,j}(t_{n+\frac{\alpha}{2}})\right) = \mathbf{V}\left(t_{n+\frac{\alpha}{2}}, \mathbf{x}_{i+1/2,j} - \frac{\alpha}{2}\mathbf{d}_{i+1/2,j}\right),$$

we write

$$\mathbf{d}_{i+1/2,j} = \Delta t \mathbf{V}\left(t_{n+\frac{\alpha}{2}}, \mathbf{x}_{i+1/2,j} - \frac{\alpha}{2}\mathbf{d}_{i+1/2,j}\right). \tag{9}$$

The velocity at $t_{n+\frac{\alpha}{2}}$ may be defined by extrapolation from the two previous time levels by the formula

$$\mathbf{V}(t_{n+\frac{\alpha}{2}}, \mathbf{x}_{i+1/2,j}) = \frac{\alpha}{2}\Big(3\mathbf{V}(t_n, \mathbf{x}_{i+1/2,j}) - \mathbf{V}(t_{n-1}, \mathbf{x}_{i+1/2,j})\Big). \tag{10}$$

Equations (9) and (10) give an implicit equation for $\mathbf{d}$ in terms of the known velocity field at two previous time levels. To compute $\mathbf{d}_{i+1/2,j}$ we consider the following successive iteration

$$\begin{aligned}
\mathbf{d}^{(0)}_{i+1/2,j} &= \alpha\Delta t\left[\frac{3}{2}\mathbf{V}\left(t_n, \mathbf{x}_{i+1/2,j}\right) - \frac{1}{2}\mathbf{V}\left(t_{n-1}, \mathbf{x}_{i+1/2,j}\right)\right], \\
\mathbf{d}^{(m)}_{i+1/2,j} &= \alpha\Delta t\left[\frac{3}{2}\mathbf{V}\left(t_n, \mathbf{x}_{i+1/2,j} - \frac{1}{2}\mathbf{d}^{(m-1)}_{i+1/2,j}\right) - \frac{1}{2}\mathbf{V}\left(t_{n-1}, \mathbf{x}_{i+1/2,j} - \frac{1}{2}\mathbf{d}^{(m-1)}_{i+1/2,j}\right)\right], \quad m = 1, 2, \ldots.
\end{aligned} \tag{11}$$

It follows from equations (10) and (11) along with a Taylor series expansion of the position vector $\mathbf{X}_{i+1/2,j}(t_n)$ that

$$\left\|\mathbf{d} - \mathbf{d}^{(m)}\right\| \leq \frac{\alpha}{4}\Delta t \left\|\nabla \mathbf{V}\right\| \left\|\mathbf{d} - \mathbf{d}^{(m-1)}\right\|, \qquad m = 1, 2, \ldots, \tag{12}$$

where $\|\cdot\|$ is the Euclidean norm in $\mathbb{R}^2$. Hence, a sufficient condition for convergence is that the velocity gradient satisfies [12]

$$\alpha\Delta t \left\|\nabla \mathbf{V}\right\| \leq 1. \tag{13}$$

We can conclude from (12) that a few iterations (2 or 3) are enough to approximate $\mathbf{d}$ up to $\mathcal{O}(\Delta t^3)$. In our computational test examples, the iterations in (11) were continued until the trajectory changed by less than 10^{-5}. However, in practice it is not recommended to repeat the iteration process more than a few times due to efficiency considerations. Notice that this low number of iterations in our approach is mainly attributed to the stability condition for the time stepping (3) which inclusively ensures the convergence condition (13).

In general $\mathbf{X}_{i+1/2,j}(t_n)$ will not coincide with the spatial position of a gridpoint. A requirement is then that the scheme to compute $\mathbf{X}_{i+1/2,j}(t_n)$ be equipped with a search-locate algorithm to find the host element where such point is located. For structured grids this step can be simple as index checking or *ad hoc* searching. Eulerian-Lagrangian methods on unstructured meshes have also been studied in [4, 9] among others. Assuming that a suitable approximation is made for $\mathbf{d}_{i+1/2,j}$ in (11), then $\mathbf{X}_{i+1/2,j}(t_n)$ would not lie on a gridpoint, so the solutions at the characteristic feet must be obtained by interpolation from known values at the gridpoints of the control volume where $\mathbf{X}_{i+1/2,j}(t_n)$ belongs. Thus, once the characteristics

$\mathbf{X}_{i+1/2,j}(t_n) = (X_{i+1/2}(t_n), Y_j(t_n))^T$ and $\mathbf{Y}_{i,j+1/2}(t_n) = (X_i(t_n), Y_{j+1/2}(t_n))^T$ are accurately calculated, the numerical fluxes in (3) are reconstructed using

$$\begin{aligned} U^n_{i+1/2,j} &= u\left(t_n + \alpha\Delta t, x_{i+1/2}, y_j\right) = \widetilde{U}\left(t_n, X_{i+1/2}(t_n), y_j\right), \\ U^n_{i,j+1/2} &= u\left(t_n + \alpha\Delta t, x_i, y_{j+1/2}\right) = \widetilde{U}\left(t_n, x_i, Y_{j+1/2}(t_n)\right), \end{aligned} \tag{14}$$

where $\widetilde{U}\left(t_n, X_{i+1/2}(t_n), y_j\right)$ and $\widetilde{U}\left(t_n, x_i, Y_{j+1/2}(t_n)\right)$ are the solutions at the characteristic feet computed by interpolation from the gridpoints of the control volume where the departure points reside *i.e.*

$$\begin{aligned} \widetilde{U}\left(t_n, X_{i+1/2}(t_n), y_j\right) &= \mathcal{P}\Big(U\left(t_n, X_{i+1/2}(t_n), y_j\right)\Big), \\ \widetilde{U}\left(t_n, x_i, Y_{j+1/2}(t_n)\right) &= \mathcal{P}\Big(U\left(t_n, x_i, Y_{j+1/2}(t_n)\right)\Big), \end{aligned} \tag{15}$$

where $\mathcal{P}$ represents an interpolating polynomial. For instance, a Lagrange-based interpolation polynomials can be formulated as

$$\begin{aligned} \mathcal{P}\Big(U\left(t_n, X_{i+1/2}(t_n), y_j\right)\Big) &= \sum_{k,l} \mathcal{L}_{k,l}(X_{i+1/2}, y_j) U^n_{k,l}, \\ \mathcal{P}\Big(U\left(t_n, x_i, Y_{j+1/2}(t_n)\right)\Big) &= \sum_{k,l} \mathcal{L}_{k,l}(x_i, Y_{j+1/2}) U^n_{k,l}, \end{aligned} \tag{16}$$

with $\mathcal{L}_{k,l}$ are the Lagrange basis polynomials given by

$$\mathcal{L}_{k,l}(x,y) = \prod_{\substack{p=0 \\ p\neq k}} \prod_{\substack{q=0 \\ q\neq l}} \frac{x - x_p}{x_k - x_p} \frac{y - y_q}{y_l - y_q}.$$

Note that the proposed finite volume scheme can be interpreted as a predictor stage (14) where the numerical fluxes $F_{i\pm1/2,j}$ and $G_{i,j\pm1/2}$ are calculated followed by a corrector stage (3) where the conservation property is preserved. It is also worth remarking that the introduction of the time parameter α in the predictor stage (14) is motivated by the fact that the time step $t_n + \alpha\Delta t$ should not be larger than the value t_{n+1} which corresponds to the time required for the fastest wave generated at the interfaces $(x_{i+1/2}, y_j)$ and $(x_i, y_{j+1/2})$ to leave the cell $\mathcal{C}_{i,j}$. In our implementation, we have used a global fixed value for α however, a local selection $\alpha^n_{i,j}$ is also possible.

3 Analysis of the Finite Volume Characteristics Method

In this section we assume a bilinear interpolating polynomial $\mathcal{P}$ is used in the predictor stage (14). Thus, we have the following results:

LEMMA **3.1** *Assume a linear interpolating polynomial $\mathcal{P}$ is used in the predictor stage* (14)*. The finite volume characteristics method is L^∞-stable.*

PROOF: Applied to the problem (1), the corrector stage (3) gives

$$U^{n+1}_{i,j} = U^n_{i,j} - \nu\left(f\left(U^n_{i+1/2,j}\right) - f\left(U^n_{i-1/2,j}\right)\right) - \eta\left(g\left(U^n_{i,j+1/2}\right) - g\left(U^n_{i,j-1/2}\right)\right), \tag{17}$$

with $\nu = \frac{\Delta t}{\Delta x}$ and $\eta = \frac{\Delta t}{\Delta y}$. The averaged states are given by

$$\begin{aligned} U^n_{i+1/2,j} &= \mathcal{P}\Big(U\left(t_n, X_{i+1/2}, Y_j\right)\Big), \\ U^n_{i,j+1/2} &= \mathcal{P}\Big(U\left(t_n, X_i, Y_{j+1/2}\right)\Big), \end{aligned} \tag{18}$$

where the characteristics $\left(X_{i+1/2}, Y_j\right)$ and $\left(X_i, Y_{j+1/2}\right)$ are given by

$$\begin{aligned} X_{i+1/2} &= x_{i+1/2} - \alpha\Delta t f'\left(U^n_{i+1/2,j}\right), \\ Y_j &= y_j - \alpha\Delta t g'\left(U^n_{i+1/2,j}\right), \end{aligned}$$

and

$$\begin{aligned} X_i &= x_i - \alpha\Delta t f'\left(U^n_{i,j+1/2}\right), \\ Y_{j+1/2} &= y_{j+1/2} - \alpha\Delta t g'\left(U^n_{i,j+1/2}\right), \end{aligned}$$

respectively. Using the bilinear interpolating polynomial, the solutions at the departure points in (18) are calculated as

$$U^n_{i+1/2,j} = \tilde{U}^n_S + \frac{Y_j - y_j}{\Delta y}\left(\tilde{U}^n_N - \tilde{U}^n_S\right) \tag{19}$$

where U^n_S and U^n_N are the south and north linear interpolated solutions in the control volume $\mathcal{C}_{i,j}$ as shown in Figure 3 *i.e.*,

$$\tilde{U}^n_S = U(t_n, X_{i+1/2}, y_j), \qquad \tilde{U}^n_N = U(t_n, X_{i+1/2}, y_{j+1}),$$

Thus,

$$\begin{aligned} \tilde{U}^n_S &= U^n_{i,j} + \left(\frac{1}{2} - \alpha\nu f'(U^n_{i+1/2,j})\right)(U^n_{i+1,j} - U^n_{i,j}), \\ \tilde{U}^n_N &= U^n_{i,j+1} + \left(\frac{1}{2} - \alpha\nu f'(U^n_{i+1/2,j})\right)\left(U^n_{i+1,j+1} - U^n_{i,j+1}\right). \end{aligned}$$

Analogously,

$$U^n_{i,j+1/2} = \tilde{U}^n_W + \frac{X_i - x_i}{\Delta x}\left(\tilde{U}^n_E - \tilde{U}^n_W\right) \tag{20}$$

where U^n_E and U^n_W are the east and west solutions illustrated in Figure 3 and defined as

$$\tilde{U}^n_W = U(t_n, x_i, Y_{j+1/2}), \qquad \tilde{U}^n_E = U(t_n, x_{i+1}, Y_{j+1/2}),$$

with

$$\begin{aligned} \tilde{U}^n_W &= U^n_{i,j} + \left(\frac{1}{2} - \alpha\eta g'(U^n_{i,j+1/2})\right)(U^n_{i,j+1} - U^n_{i,j}), \\ \tilde{U}^n_E &= U^n_{i+1,j} + \left(\frac{1}{2} - \alpha\eta g'(U^n_{i,j+1/2})\right)\left(U^n_{i+1,j+1} - U^n_{i+1,j}\right). \end{aligned}$$

Note that we have assumed that $f' < 0$ and $g' < 0$. By construction the problem (14) has a unique solution. There exist $\beta^n_{i,j} \in [U^n_{i-1/2,j}, U^n_{i+1/2,j}]$ and $\gamma^n_{i,j} \in [U^n_{i,j-1/2}, U^n_{i,j+1/2}]$ such that

$$\begin{aligned} f(U^n_{i+1/2,j}) - f(U^n_{i-1/2,j}) &= f'(\beta^n_{i,j})\left(U^n_{i+1/2,j} - U^n_{i-1/2,j}\right), \\ g(U^n_{i,j+1/2}) - g(U^n_{i,j-1/2}) &= g'(\gamma^n_{i,j})\left(U^n_{i,j+1/2} - U^n_{i,j-1/2}\right), \end{aligned}$$

Thus, substituting (19) and (20) in the corrector stage (17) we obtain

$$U^{n+1}_{i,j} = U^n_{i,j} - \nu f'(\beta^n_{i,j})\left[U^n_{i+1/2,j} - U^n_{i-1/2,j}\right] - \eta g'(\gamma^n_{i,j})\left[U^n_{i,j+1/2} - U^n_{i,j-1/2}\right]. \tag{21}$$

From (19) we can write

$$\min(U^n_N, U^n_S) \le U^n_{i+1/2,j} \le \max(U^n_N, U^n_S), \tag{22}$$

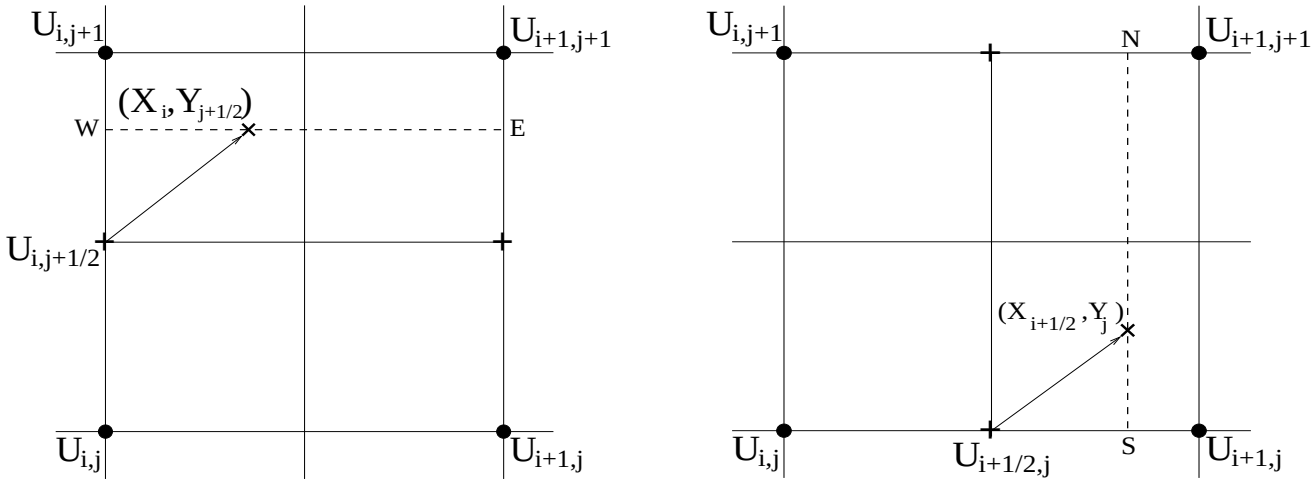

Figure 3: Mesh points used in the bilinear interpolation procedure.

where

$$\begin{aligned} \min(U^n_{i,j+1}, U^n_{i+1,j+1}) \quad &\leq U^n_N \leq \quad \max(U^n_{i,j+1}, U^n_{i+1,j+1}), \\ \min(U^n_{i,j}, U^n_{i+1,j}) \quad &\leq U^n_S \leq \quad \max(U^n_{i,j}, U^n_{i+1,j}). \end{aligned} \tag{23}$$

Hence, (22) and (23) lead to

$$\min(U^n_{i,j}, U^n_{i,j+1}, U^n_{i+1,j}, U^n_{i+1,j+1}) \leq U^n_{i+1/2,j} \leq \max(U^n_{i,j}, U^n_{i,j+1}, U^n_{i+1,j}, U^n_{i+1,j+1}).$$

In a similar manner

$$\min(U^n_E, U^n_W) \leq U^n_{i,j+1/2} \leq \max(U^n_E, U^n_W), \tag{24}$$

where

$$\begin{aligned} \min(U^n_{i+1,j}, U^n_{i+1,j+1}) \quad &\leq U^n_E \leq \quad \max(U^n_{i+1,j}, U^n_{i+1,j+1}), \\ \min(U^n_{i,j}, U^n_{i,j+1}) \quad &\leq U^n_W \leq \quad \max(U^n_{i,j}, U^n_{i,j+1}). \end{aligned} \tag{25}$$

Hence (24) and (25) lead to

$$\min(U^n_{i,j}, U^n_{i,j+1}, U^n_{i+1,j}, U^n_{i+1,j+1}) \leq U^n_{i,j+1/2} \leq \max(U^n_{i,j}, U^n_{i,j+1}, U^n_{i+1,j}, U^n_{i+1,j+1}) \tag{26}$$

From (21) we can write

$$\begin{aligned} \left|U^{n+1}_{i,j}\right| &\leq \left[1 + 2\left(\nu|f'(\beta^n_{i,j})| + \eta|g'(\gamma^n_{i,j})|\right)\right] ||U^n||_\infty, \qquad \forall (i,j) \in \mathbb{Z}, \\ \left\|U^{n+1}\right\|_\infty &\leq [1 + \omega\Delta t]\, ||U^n||_\infty, \\ &\leq (1 + \omega\Delta t)^n\, ||U^0||_\infty, \\ &\leq e^{n\omega\Delta t} ||U^0||_\infty, \\ &\leq e^{\omega T} ||U^0||_\infty, \qquad \forall t \in [0, T], \end{aligned}$$

where

$$\omega = 2\left(\frac{1}{\Delta x}\left\|f'(\beta^n)\right\|_\infty + \frac{1}{\Delta y}\left\|g'(\gamma^n)\right\|_\infty\right).$$

Therefore, the characteristic finite volume scheme is L^∞-stable.

■

4 Extension to Convection-Diffusion Problems

In the current study, we are concerned with nonlinear convection-diffusion problems where the convection dominates the diffusion. The mathematical formulation of the problem reads

$$\frac{\partial u}{\partial t}+\frac{\partial f(u)}{\partial x}+\frac{\partial g(u)}{\partial y}-\nu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)=q, \tag{27}$$

where ν is the diffusion (viscosity) coefficient and q is the source term. Applied to the equation (27), the finite volume characteristics method yields

$$\begin{aligned} U_{i,j}^{n+1} &= U_{i,j}^{n}-\frac{\Delta t}{\Delta x}\left(F_{i+1/2,j}^{n}-F_{i-1/2,j}^{n}\right)-\frac{\Delta t}{\Delta y}\left(G_{i,j+1/2}^{n}-G_{i,j-1/2}^{n}\right), \\ &\quad -\Delta t\Big(Q_{i,j}^{n}-\nu\mathcal{D}_x^2 U_{i,j}^{n}-\nu\mathcal{D}_y^2 U_{i,j}^{n}\Big), \end{aligned} \tag{28}$$

where $F_{i\pm1/2,j}^{n}$ and $G_{i,j\pm1/2}^{n}$ are the numerical fluxes calculated using the same Lagrangian stage described in section 2.2. In (28), $Q_{i,j}^{n}$ is the discretization of the source term q in (27), $\mathcal{D}_x^2$ and $\mathcal{D}_y^2$ are difference notations defined as

$$\mathcal{D}_x^2 U_{i,j}^{n}=\frac{U_{i+1,j}^{n}-2U_{i,j}^{n}+U_{i-1,j}^{n}}{(\Delta x)^2}, \qquad \mathcal{D}_y^2=\frac{U_{i,j+1}^{n}-2U_{i,j}^{n}+U_{i,j-1}^{n}}{(\Delta y)^2}. \tag{29}$$

To summarize, the implementation of our new finite volume Eulerian-Lagrangian algorithm to solve the convection-diffusion problem (27) is carried out in the following steps:

Step 1. *Compute the departure points* $\mathbf{X}_{i+1/2,j}(t_n)$ *and* $\mathbf{Y}_{i,j+1/2}(t_n)$ *using the iterative procedure* (11).

Step 2. *Compute the intermediate states* $U_{i+1/2,j}^{n}$ *and* $U_{i,j+1/2}^{n}$

$$U_{i+1/2,j}^{n}=\widetilde{U}\left(t_n+\alpha\Delta t,\mathbf{X}_{i+1/2,j}(t_n)\right) \qquad \text{and} \qquad U_{i,j+1/2}^{n}=\widetilde{U}\left(t_n+\alpha\Delta t,\mathbf{Y}_{i,j+1/2}(t_n)\right),$$

employing an interpolation procedure.

Step 3. *Evaluate the numerical fluxes* $F_{i+1/2,j}^{n}=f\left(U_{i+1/2,j}^{n}\right)$ *and* $G_{i,j+1/2}^{n}=g\left(U_{i,j+1/2}^{n}\right)$.

Step 4. *Evaluate the difference quotients* $\mathcal{D}_x^2$ *and* $\mathcal{D}_y^2$ *using* (29).

Step 5. *update the numerical solution* $U_{i,j}^{n+1}$ *using* (28).

Note that other interpolation procedures such as Spline or Hermite interpolation methods or interpolation techniques based on radial basis functions can also be applied in step 2.

5 Numerical Results

In this section, we perform some numerical tests with our finite volume Eulerian-Lagrangian scheme. In all our computations a fixed courant number $\mathrm{Cr}=0.8$ is used while the time step Δt is varied according to the stability condition

$$\Delta t=\mathrm{Cr}\min\left(\frac{h}{\lambda\sqrt{2\alpha}},\frac{1}{\lambda\alpha},\frac{h^2}{\nu}\right),$$

where $\lambda=\max\Big(\left|f'(u)\right|,\left|g'(u)\right|\Big)$ and $h=\max\left(\Delta x,\Delta y\right)$. In all results presented in this section the time parameter $\alpha=\frac{1}{2}$ and bilinear interpolation procedure is used in the predictor stage unless stated. The

selection of the parameter α is made based on the analysis reported in [2] for one-dimensional problems. We first of all perform accuracy tests on a linear problem. Thereafter, we consider two test examples using the Burgers equation and we also show numerical results for the Buckley-Leverett equation. For comparison, we compare the results obtained using our Finite Volume Characteristics (FVC) method to those obtained using the well established Roe and Rusanov schemes, and also using the conventional semi-Lagrangian (SLAG) method. For completeness, the formulation of Roe, Rusanov and SLAG methods is briefly described in the Appendix.

5.1 Advection-diffusion problem

This example considers the advection-diffusion of a rotating Gaussian pulse widely used in the literature to test the accuracy of semi-Lagrangian methods. The equations are of the form (27) with flux functions $f(u) = -4yu$, $g(u) = 4xu$ and the source term $q = 0$. Initial and boundary conditions are taken from the analytical solution

$$u(t, x, y) = \frac{\sigma^2}{\sigma^2 + 4\nu t} \exp\left(-\frac{(\bar{x} - x_0)^2 + (\bar{y} - y_0)^2}{\sigma^2 + 4\nu t}\right),$$

where $\bar{x} = x\cos(4t) + y\sin(4t)$, $\bar{y} = -x\sin(4t) + y\cos(4t)$, $x_0 = 0.5$, $y_0 = 0.75$ and $\sigma^2 = 0.002$. The computational domain is $[0, 1] \times [0, 1]$ covered by uniform mesh with $h = \Delta x = \Delta y$, and the time period required for one complete rotation is $\frac{\pi}{2}$. The purpose of this test example is to compare the FVC results with those computed using the conventional SLAG method.

First we consider the pure advection test example corresponding to $\nu = 0$. Figure 4 represents the computed results using the FVC scheme after 1 revolution and 3 revolutions on a mesh of 50×50 gridpoints. In Figure 5 we display the associated 10 equi-distributed contourlines of the solutions. For a comparison reason, we have also included in these figures the computational resulted obtained using the conventional SLAG scheme and the analytical solutions. The one-dimensional plots in Figure 6 correspond to a cross section at $y = 0.75$ of the results obtained after 1 revolution and 3 revolutions. A visual comparison of the results in these figures shows severe numerical dissipation, overshoot, deformation and phase errors in the SLAG solutions. After 3 revolutions, the SLAG method exhibits nonphysical oscillations and substantially greater distortion, specially at the feet of the Gaussian pulse where the gradient is sharper. From the same figures we observe an absence of these oscillations in the FVC results. It is evident that, after one revolution, both methods give roughly similar results with some small differences on the maximum value of the numerical solutions. However, by increasing the number of revolutions to 3, the FVC results are more accurate than those of the conventional SLAG method. It is clear that the FVC scheme performs best for this test example. It should be pointed out that rotating the Gaussian pulse for more than 3 revolutions results in nonphysical solutions for the SLAG method whereas the FVC scheme still produces satisfactory results.

Next we include the physical diffusion in the problem by solving the advection-diffusion equation with a diffusion coefficient $\nu = 10^{-3}$. The obtained results using the FVC and SLAG schemes are presented in Figure 7 for the solution profiles and in Figure 8 for the solution contours. Figure 9 shows the cross sections at $y = 0.75$ of the computed solutions after 1 and 3 revolutions. It is clear from the results presented that the numerical diffusion is more pronounced in the results obtained using the SLAG scheme and a larger dispersion errors have been detected in the SLAG results, compare the solution contours in Figure 8. As can be seen from the cross sections in Figure 9, after one revolution using a mesh of 100×100 gridpoints, both FVC and SLAG schemes give roughly similar results with some small differences on the maximum value of the numerical solutions. However, by decreasing the number of gridpoints to 50×50 or increasing the number revolutions to 3, the FVC results are more accurate than those of the conventional SLAG method. Again the FVC scheme performs best for this test example of linear advection-diffusion problems.

A quantitative comparison of the results computed by FVC and SLAG methods for different number of revolutions is given in Table 1 for $\nu = 0$ and in Table 2 for $\nu = 10^{-3}$. We report the L^1 errors, the relative

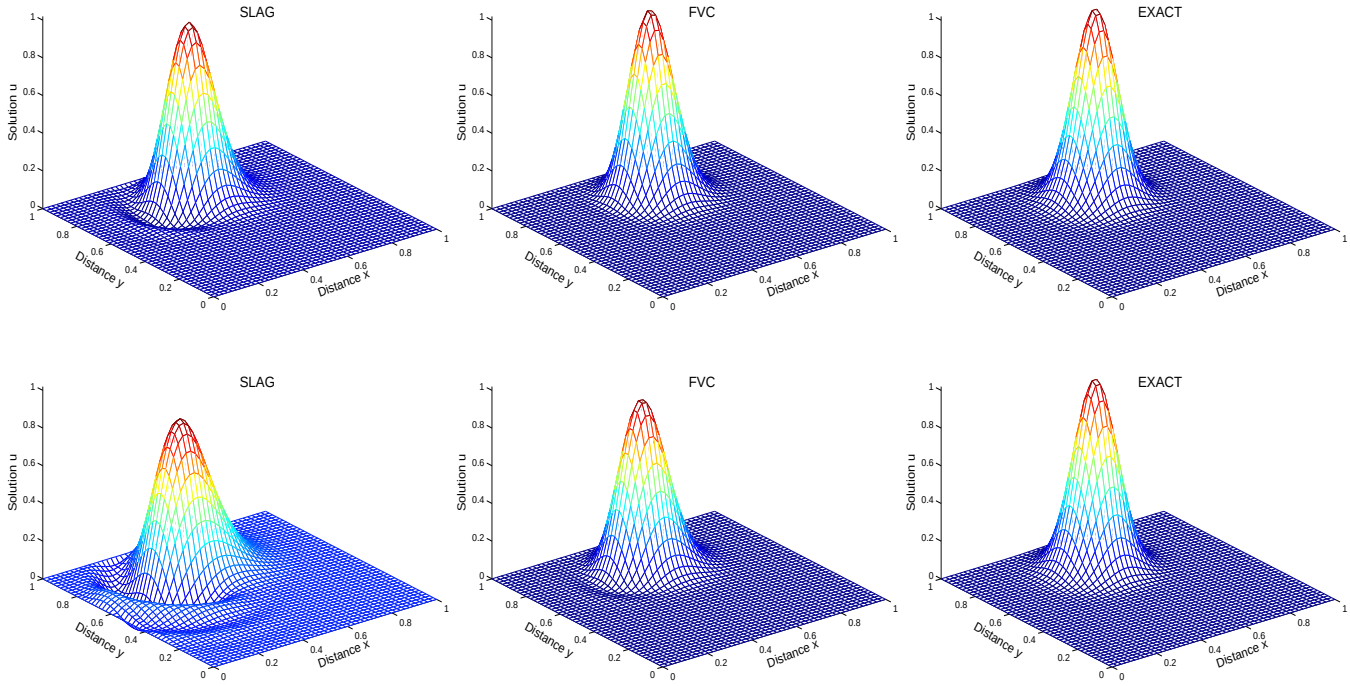

Figure 4: Results for the pure advection problem after 1 revolution (first row) and 3 revolution (second row) using 50×50 gridpoints.

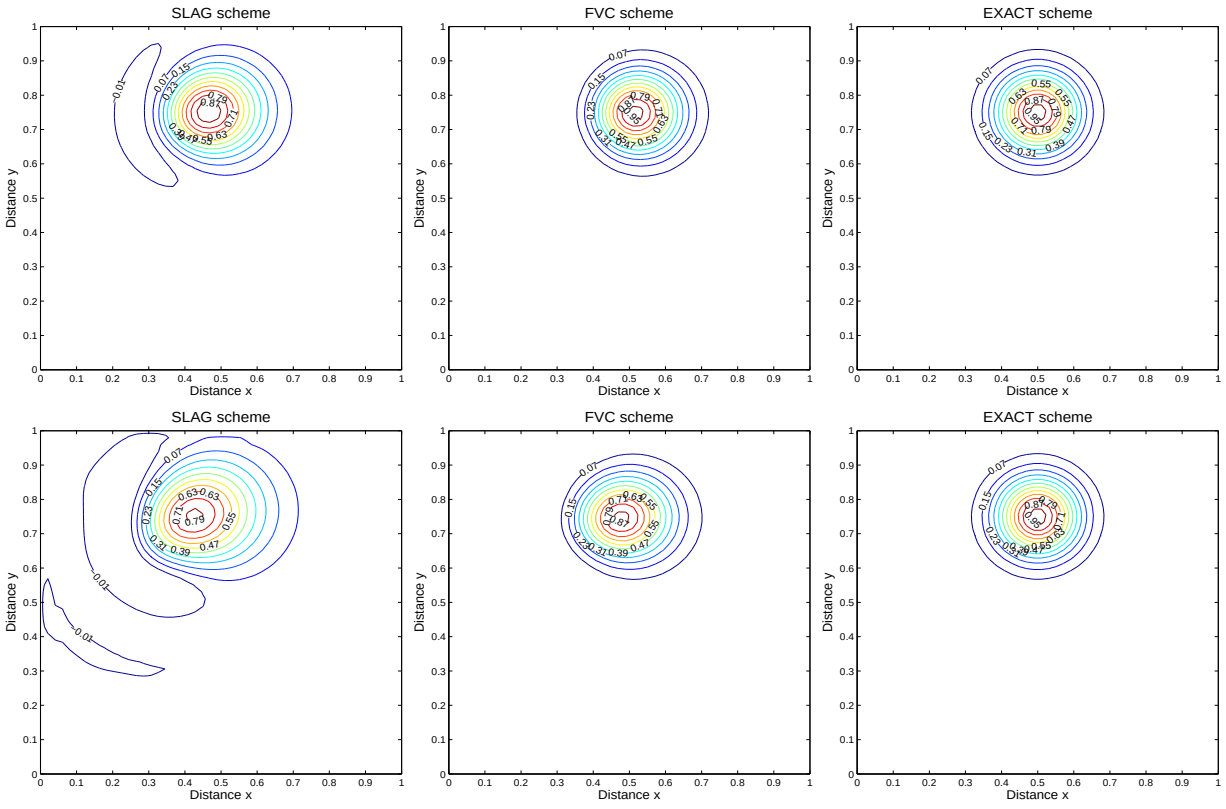

Figure 5: Results for the pure advection problem after 1 revolution (first row) and 3 revolution (second row) using 50×50 gridpoints.

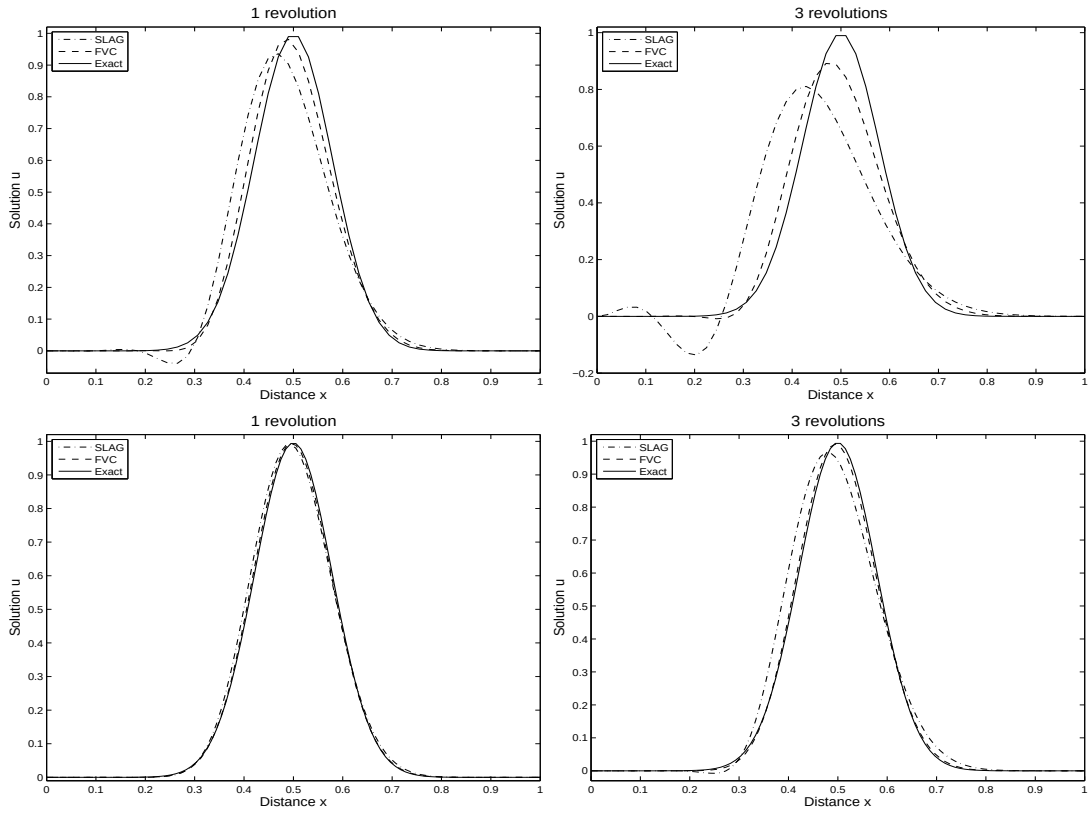

Figure 6: Cross sections at $x = 0.75$ of the results for the pure advection problem after 1 revolution (left) and 3 revolutions (right) using 50×50 gridpoints (top) and 100×100 gridpoints (bottom).

Table 1: Results for advection of the Gaussian pulse test after 1 and 3 revolutions using $\nu = 0$. The analytical minimum and maximum are 0 and 1, respectively. Here the CPU times are given in seconds.

		After 1 revolution				
Scheme	Mesh	min	max	r_{mass}	L^1-error	CPU
FVC	50×50	-0.00000820	0.98111	0.9991	0.00312815	4.03
	100×100	-0.00000247	0.99762	0.9994	0.00071643	11.07
	200×200	-0.00000213	0.99933	0.9995	0.00017851	60.34
SLAG	50×50	-0.0434479	0.93585	1.0392	0.0100468	2.06
	100×100	-0.00029908	0.99102	1.0393	0.0030806	5.64
	200×200	-0.00023464	0.99875	1.0394	0.0023233	30.72

		After 3 revolutions				
Scheme	Mesh	min	max	r_{mass}	L^1-error	CPU
FVC	50×50	-0.00057105	0.92832	0.99927	0.00952134	10.75
	100×100	-0.00001454	0.9916	0.99929	0.00186891	30.08
	200×200	-0.00000905	0.99857	0.99938	0.00050257	167.01
SLAG	50×50	-0.14383783	0.81044	1.1228	0.03040275	5.46
	100×100	-0.00856685	0.96704	1.1229	0.00935681	15.74
	200×200	-0.00046101	0.99672	1.1229	0.00716476	88.95

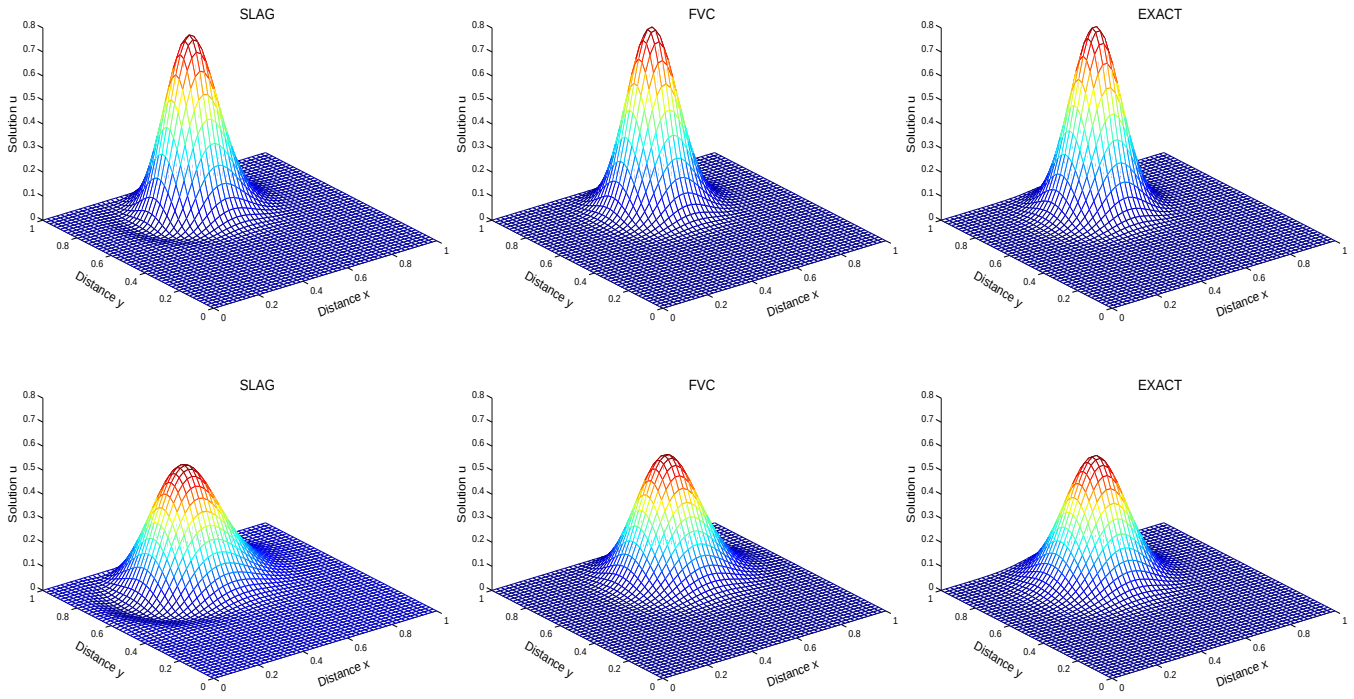

Figure 7: Results for the advection-diffusion problem with $\nu = 10^{-3}$after 1 revolution (first row) and 3 revolution (second row) using 50×50 gridpoints.

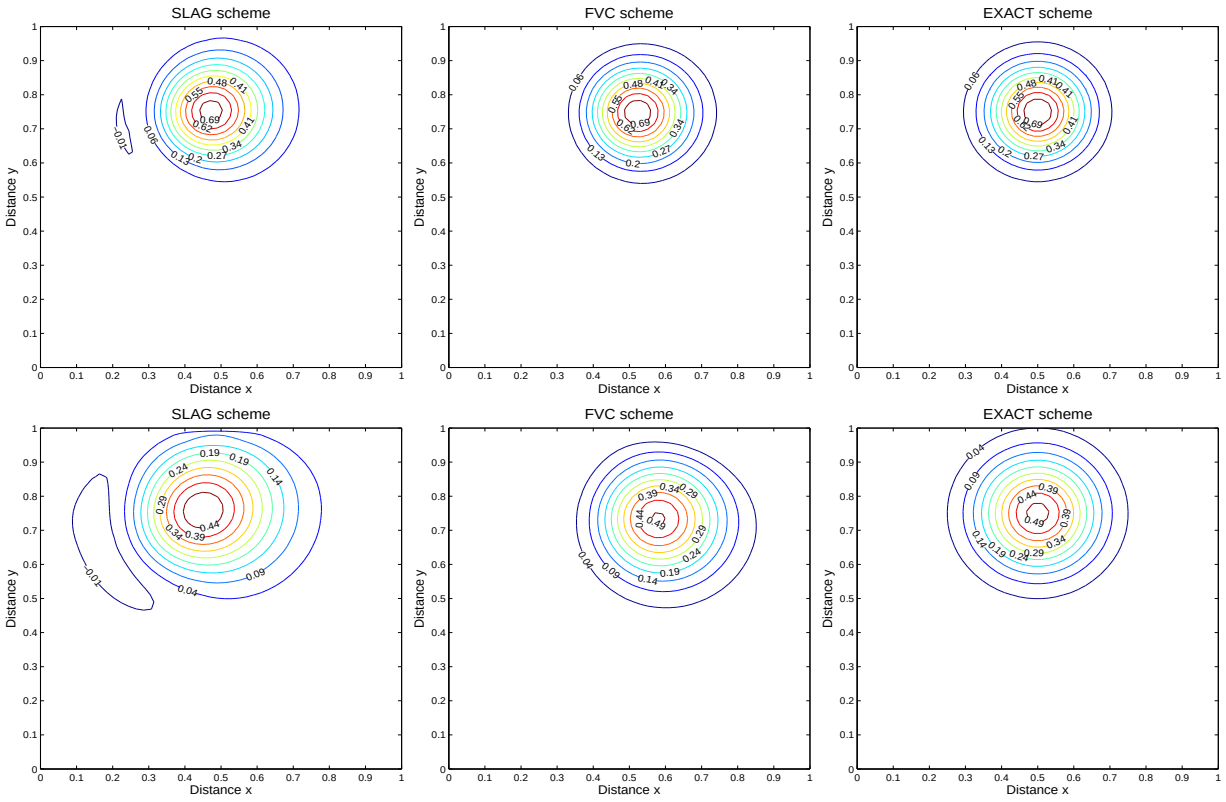

Figure 8: Results for the advection-diffusion problem with $\nu = 10^{-3}$after 1 revolution (first row) and 3 revolution (second row) using 50×50 gridpoints.

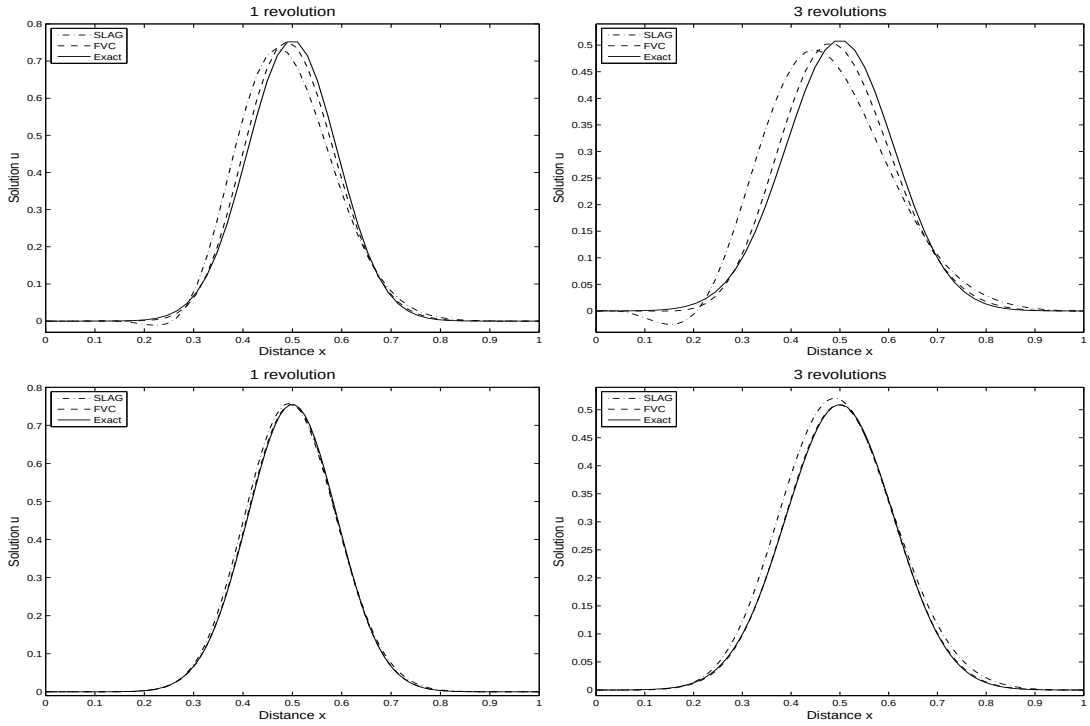

Figure 9: Cross sections at $x = 0.75$ of the results for the advection-diffusion problem with $\nu = 10^{-3}$ after 1 revolution (left) and 3 revolutions (right) using 50×50 gridpoints (top) and 100×100 gridpoints (bottom).

Table 2: Results for advection of the Gaussian pulse test after 1 and 3 revolutions using $\nu = 10^{-3}$. The analytical maximum after 1 revolution is 0.751 and after 3 revolution is 0.51. The analytical minimum is 0. Here the CPU times are given in seconds.

		After 1 revolution				
Scheme	Mesh	min	max	r_{mass}	L^1-error	CPU
FVC	50×50	-0.00000527	0.75306	0.99221	0.00206861	5.02
	100×100	-0.00000178	0.7569	0.99333	0.00048586	16.96
	200×200	-0.00000063	0.75747	0.99394	0.00010073	86.01
SLAG	50×50	-0.01231785	0.73542	1.0332	0.00668748	2.93
	100×100	-0.00027517	0.75822	1.0331	0.00241946	9.12
	200×200	-0.00006934	0.76081	1.033	0.00200973	45.3

		After 3 revolutions				
Scheme	Mesh	min	max	r_{mass}	L^1-error	CPU
FVC	50×50	-0.00001072	0.50324	0.96563	0.0035162	14.16
	100×100	-0.00000388	0.50979	0.96886	0.00081374	49.17
	200×200	-0.00000095	0.51026	0.97065	0.00019481	250.41
SLAG	50×50	-0.02678958	0.49184	1.0803	0.01145755	7.94
	100×100	-0.00083346	0.5217	1.0791	0.00518927	25.98
	200×200	-0.00022091	0.52441	1.0783	0.00478164	131.48

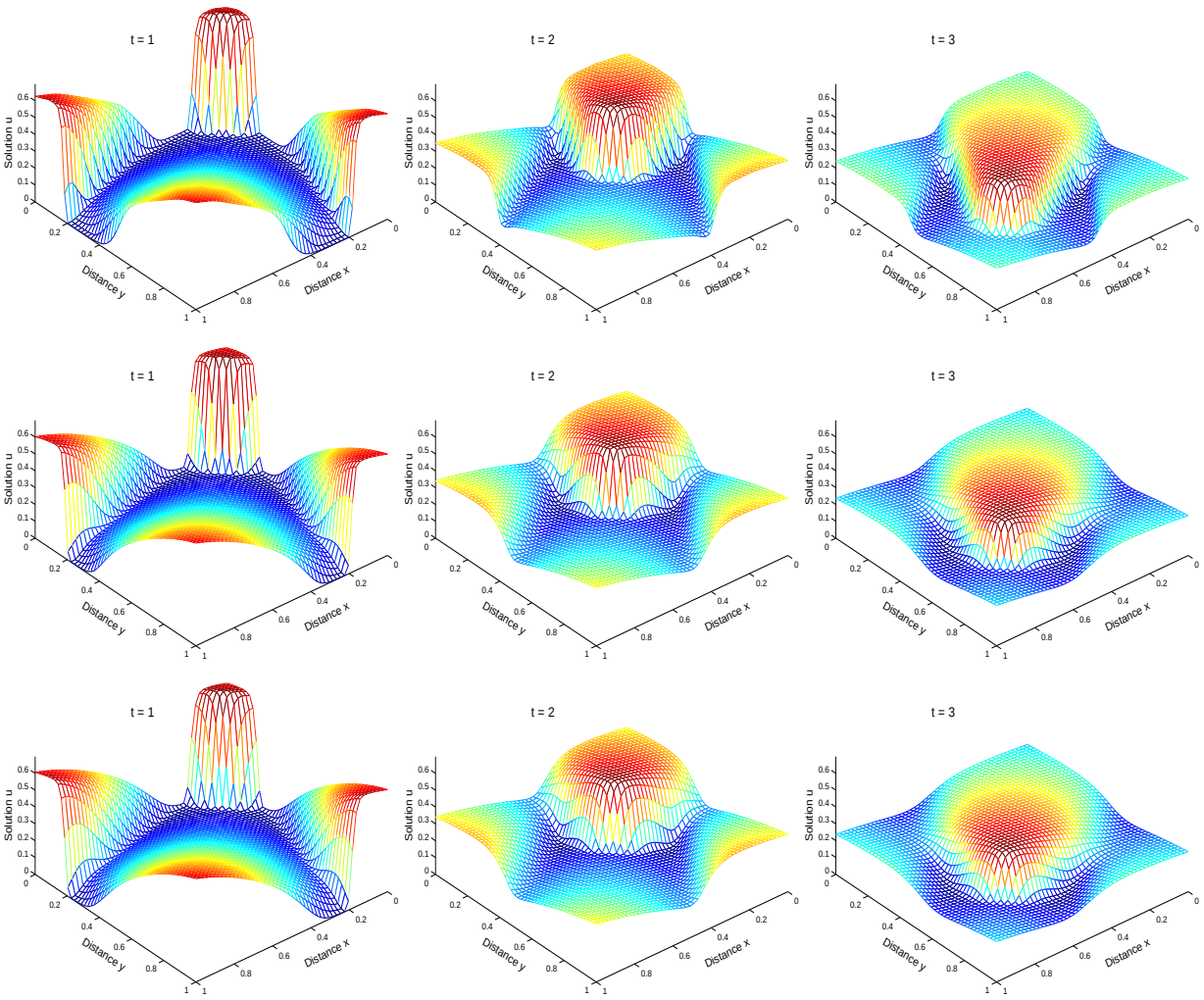

Figure 10: Results for the inviscid Burgers equation at different times using the FVC scheme (first row), Roe scheme (second row) and Rusanov scheme (third row) on a mesh of 50×50 gridpoints.

mass (*mass*), the minimum (min) and maximum (max) values of the computed solutions, and the CPU times given in seconds. We present numerical results after 1 and 3 revolutions using different meshes. In terms of the L^1 errors the FVC results are more accurate than the results obtained using the conventional SLAG method for both diffusion coefficients considered. From the values of max and min in Table 1 for $\nu = 0$ and in Table 2 for $\nu = 10^{-3}$ we observe high and negative values for the conventional SLAG results that are avoided in the FVC results. Concerning the mass conservation, Table 1 shows that, for a mesh of 100×100 gridpoints, the conventional SLAG method lost more than 12% of the initial mass after 3 rotations whereas the FVC method is mass conserving at the machine precision. It is also evident that the CPU times of the FVC method are larger than the CPU time of the conventional SLAG method. For the considered parameters, the CPU time of the FVC method is less than two times larger than the CPU time of the conventional SLAG method. It is to be remarked that, the conventional SLAG method is typically built to solve this class of convection-dominated advection-diffusion problems using times steps four to five times larger than its Eulerian counterparts.

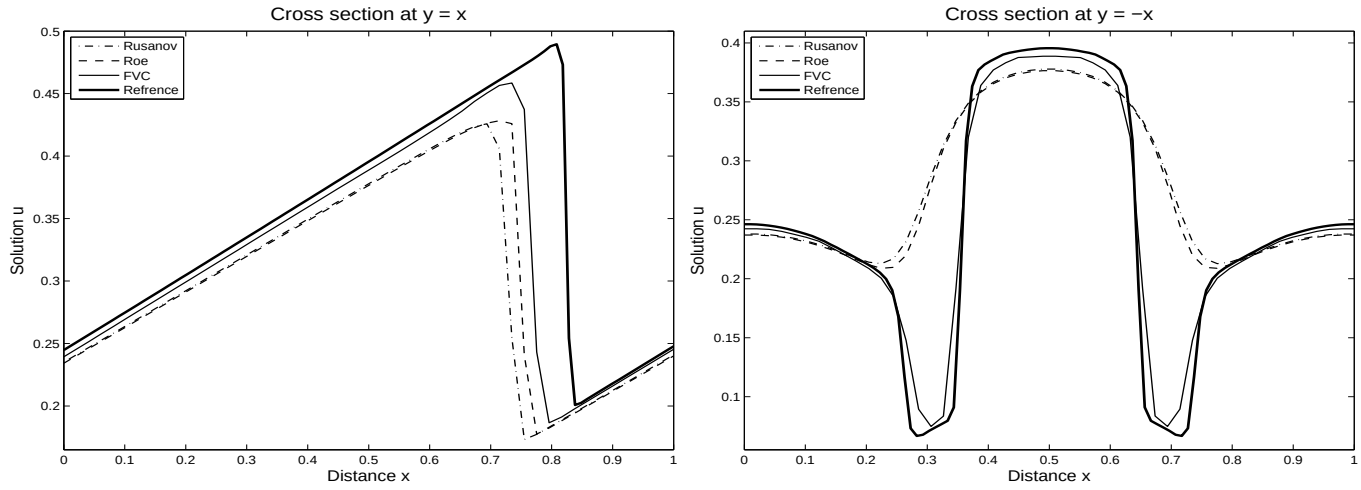

Figure 11: Cross sections at $y = x$ (left) and $y = -x$ (right) of the results for the inviscid Burgers problem at $t = 3$ using a mesh of 50×50 gridpoints.

5.2 Burgers problem

In this example we consider both the inviscid and viscous Burgers equations. We first consider the inviscid Burgers equation modelled by the conservation law (1) with nonlinear flux functions given by $f(u) = g(u) = \frac{u^2}{2}$. The problem is solved in the computational domain $[0, 1] \times [0, 1]$ augmented with periodic boundary conditions and subject to the following initial condition

$$u(0, x, y) = \sin^2(\pi x) \sin^2(\pi y), \quad (x, y) \in [0, 1] \times [0, 1].$$

It should be stressed that the unique entropy solution of this problem is smooth up to a critical time after which the solution develops a shock propagating along the diagonal in the computational domain. Notice that the conventional SLAG methods fail to resolve shocks since these methods are applied directly to the advective form of the problem under study. Therefore, the SLAG results are discarded in this example and for comparison reasons we include numerical results obtained using the Rusanov and Roe schemes. The obtained results are shown in Figure 10 at three different times, $t = 1$, 2 and 3 using a mesh of 50×50 gridpoints. It is clear that the FVC scheme accurately captures the shock and its propagation along the diagonal. However, due to the numerical dissipation, the resolved shock has been smeared out in the results obtained using the Rusanov and Roe methods. As expected, the numerical results obtained by the Roe scheme are more diffusive than those computed using the FVC scheme and the Rusanov scheme is the most diffusive. To further visualize this effect we display in Figure 11 the cross sections along the main diagonals for the results at time $t = 3$ on the mesh of 50×50 gridpoints. We also include in these figures a reference solution computed using the FVC method on a fine mesh of 500×500 gridpoints. It is clear that the shock resolution and location are deteriorated with the excessive dissipation included by the Rusanov and Roe schemes. On the other hand, the FVC solutions are completely free of spurious oscillations and the shocks are well resolved by the FVC scheme without requiring any Riemann-problem solver.

Now we turn our attention to a viscous Burgers equation which evolves to a highly convective steady state. The governing equation is

$$\frac{\partial u}{\partial t} + \lambda y(u - \frac{1}{2})\frac{\partial u}{\partial x} + \lambda x(u - \frac{1}{2})\frac{\partial u}{\partial y} - \Delta u = 0, \tag{30}$$

where λ is a constant controlling the magnitude of the nonlinear convective term, see for instance [7] for

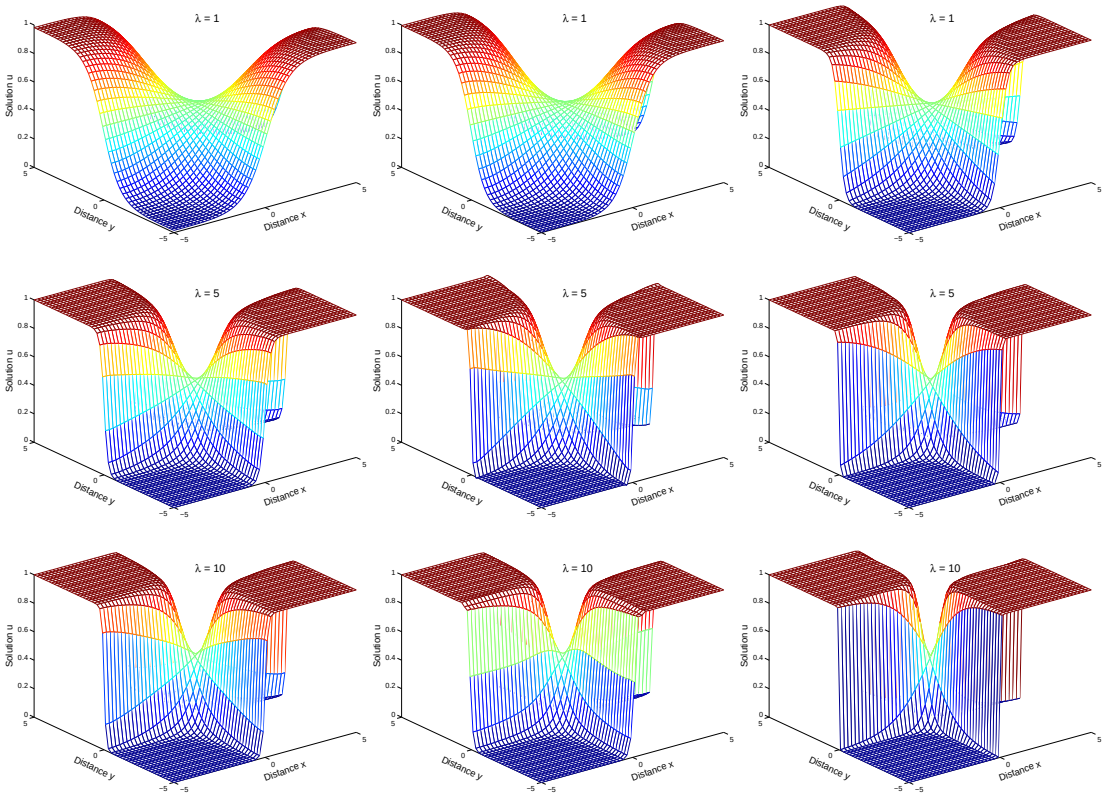

Figure 12: Results for the viscous Burgers equation at different values of λ using the SLAG scheme (first column), FVC scheme (second column) and exact solution (third column) on a mesh of 50×50 gridpoints.

further details. The boundary conditions are of Dirichlet type given by the exact steady-state solution

$$u(x,y) = \frac{1}{2}\left(1 - \tanh\left(\frac{\lambda xy}{2}\right)\right).$$

The computational domain is the square $\Omega = [-5,5] \times [-5,5]$. To define initial conditions for this problem we first divide the domain into four equally subsquares as: $\Omega_1 = I_l \times I_l$, $\Omega_2 = I_l \times I_r$, $\Omega_3 = I_r \times I_l$ and $\Omega_4 = I_r \times I_r$, with $I_l = [-5,0]$ and $I_r = [0,5]$. Then, the solution is alternated between these subsquares as follows:

$$u(0,x,y) = \begin{cases} 0, & \text{if} \quad (x,y) \in \Omega_1 \cup \Omega_4, \\ 1, & \text{if} \quad (x,y) \in \Omega_2 \cup \Omega_3, \\ \frac{1}{2}, & \text{if} \quad x = 0 \quad \text{or} \quad y = 0. \end{cases}$$

We used the new FVC method to compute the steady-state solutions for three different values of λ namely, $\lambda = 1$, $\lambda = 5$ and $\lambda = 10$. Note that since the velocity field in this example is divergence-free, the conservation form of the equation (30) is given by (27) where the flux functions depend on the solution itself and the spatial coordinates as well *i.e.*, $f(u) = \lambda y(u - \frac{1}{2})u$ and $g(u) = \lambda x(u - \frac{1}{2})u$. This test example has been solved in [7] using the conventional SLAG method and therefore, these results are compared to those obtained using our FVC scheme.

Figure 12 illustrates the obtained steady-state FVC and SLAG results and the analytical steady-state solutions using a mesh of 50×50 gridpoints. In Figure 13 we plot 10 equi-distributed contours of the solutions. It is clear that, by increasing the values of λ the convective terms become larger and steep boundary layers are formed near the vicinity of center lines in the computational domain. For low values

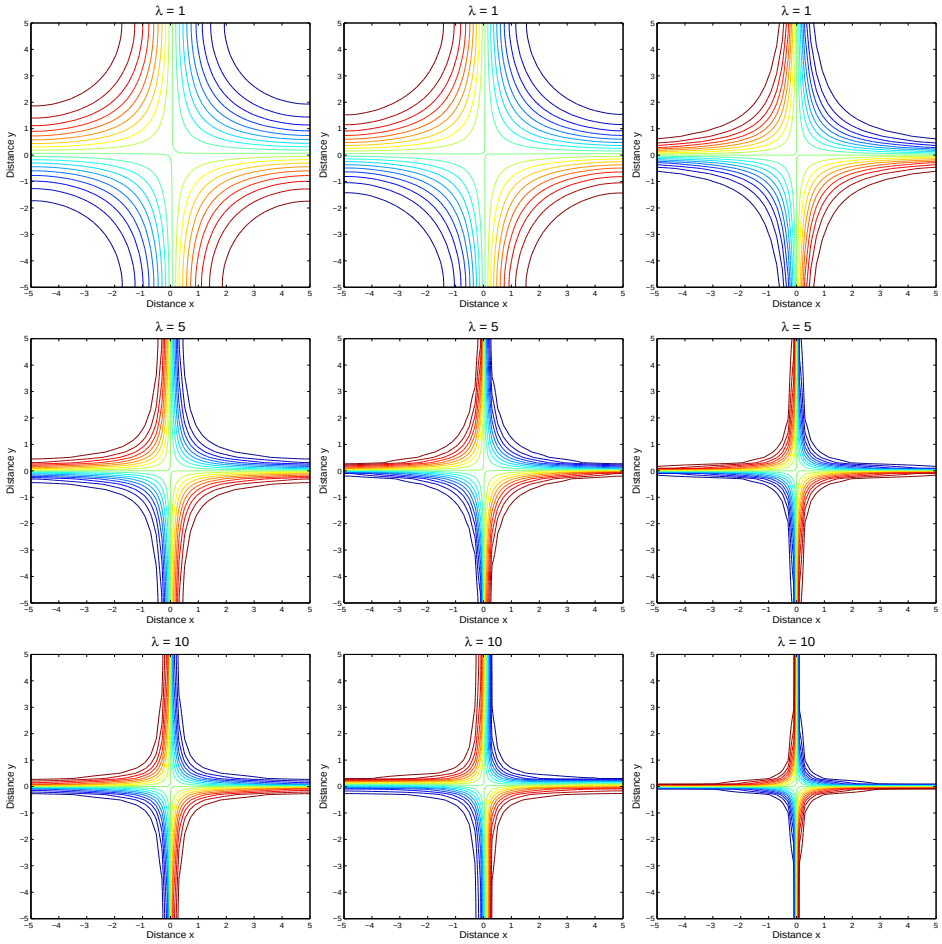

Figure 13: Results for the viscous Burgers equation at different values of λ using the SLAG scheme (first column), FVC scheme (second column) and exact solution (third column) on a mesh of 50×50 gridpoints.

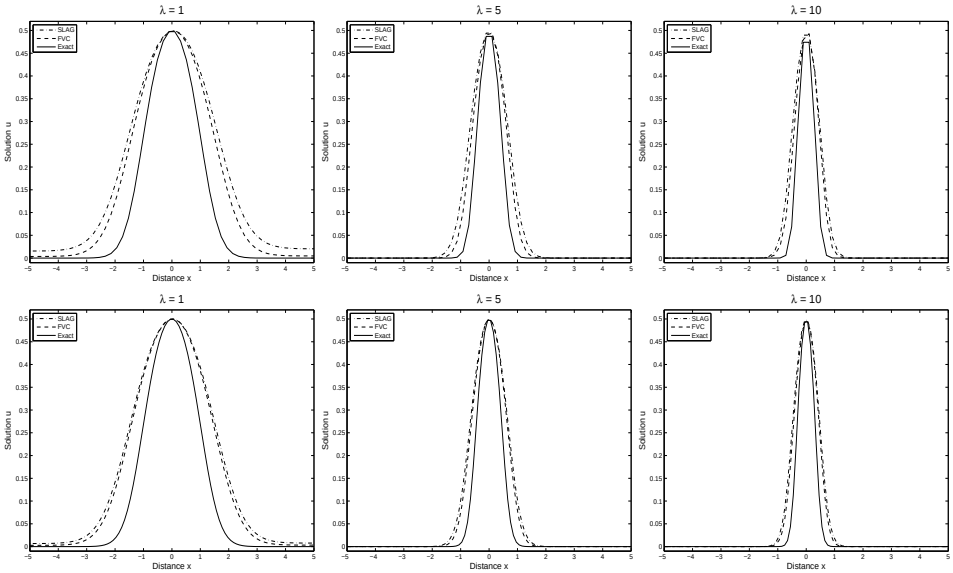

Figure 14: Cross sections at $y = x$ of the results for the viscous Burgers problem using 50×50 gridpoints (first row) and 200×200 gridpoints (second row).

of λ, the boundary layers are wide and diffuse in the flow domain. As λ increases, the boundary layers concentrate and move towards the domain center. It is apparent that the solution structures are in good agreement with the previous work in [7]. These plots give a clear view of the overall flow pattern and the effect of the convection control parameter λ on the structure of steady boundary layers in the cavity. It is worth remarking that the thinning of the boundary layers with increasing λ is evident from these plots, although the rate of this thinning is slower for the SLAG method than for the FVC method. These features clearly demonstrate the high accuracy achieved by the proposed FVC method for solving viscous Burgers problems at steady-state regimes. In addition, compared to the results published for example in [7], it can be seen that our FVC method resolves accurately the solution features and the boundary layers seem to be localized in the correct place in the flow domain.

For visualizing the comparisons, we display in Figure 14 cross sections at the main diagonal using a mesh with 50×50 gridpoints and 200×200 gridpoints. For $\lambda = 10$, it is clear that the SLAG and FVC methods produce practically identical results on the mesh of 200×200 nodes. This can be attributed to the small physical diffusion presented in the problem. However, decreasing the value of λ to 5 or 1 the results computed by FVC method are more accurate than those computed by the SLAG method. Apparently, by using the FVC method, high resolution is achieved in those regions where the flow gradients are steep such as the moving fronts. Comparing the results obtained using the considered methods, it is clear that the SLAG method produces diffusive solutions resulting in smearing the shocks. On the other hand, this numerical diffusion has remarkably been reduced in the results computed using the FVC method. Needless to say that for convection-dominated situation, the FVC method does not diffuse the fronts or gives spurious oscillations near the steep gradients. Our FVC scheme accurately approximates the solution to this steady-state problem. The results shown here compare favorably with those published in the literature for the viscous Burgers problems, see for instance [7].

5.3 Buckley-Leverett problem

We apply the finite volume Eulerian-Lagrangian scheme to the two-dimensional Buckley-Leverett given by the equation (27) with the flux functions

$$f(u) = \frac{u^2}{u^2 + (1-u)^2} \qquad \text{and} \qquad g(u) = f(u)\Big(1 - 5(1-u)^2\Big).$$

Note that the gravitational effects are included in the y-direction for this problem. We solve the problem on the space domain $[-1.5, 1.5] \times [-1.5, 1.5]$ using $\nu = 0.01$. The initial condition is given by

$$u(0, x, y) = \begin{cases} 1, & \text{if} \quad x^2 + y^2 < 0.5, \\ 0, & \text{otherwise.} \end{cases}$$

The obtained results at $t = 0.5$ on two different meshes of 50×50 and 200×200 gridpoints. Figure 15 depicts 20 equi-distributed contours of the numerical solutions. For comparison we have also included in this figure the results obtained using the Rusanov and Roe schemes. It can be clearly seen that the complicated solution structures are being captured by the FVC method. On the mesh with 50×50 gridpoints, the computed results for the Rusanov and Roe schemes are more diffusive than those computed using the FVC scheme. Increasing the mesh density to 200×200 gridpoints results in an increase of the numerical resolution of all considered schemes but the FVC method is still superior to Rusanov and Roe schemes, compare the one-dimensional profiles of the solutions at $y = 0.5$ in Figure 16. Compared with the numerical results obtained using the Rusanov and Roe schemes, the FVC scheme solves the problem accurately with less numerical diffusion than the Rusanov and Roe methods.

Note that in general, the Rusanov and Roe schemes require a solver for the Riemann problem at each time step to reconstruct the numerical fluxes, which is completely avoided in our FVC scheme. It should be pointed out that the performance of the FVC method is very attractive since the computed solution remains stable and accurate even when coarse meshes are used without requiring Riemann solvers or complicated techniques to reconstruct the numerical fluxes.

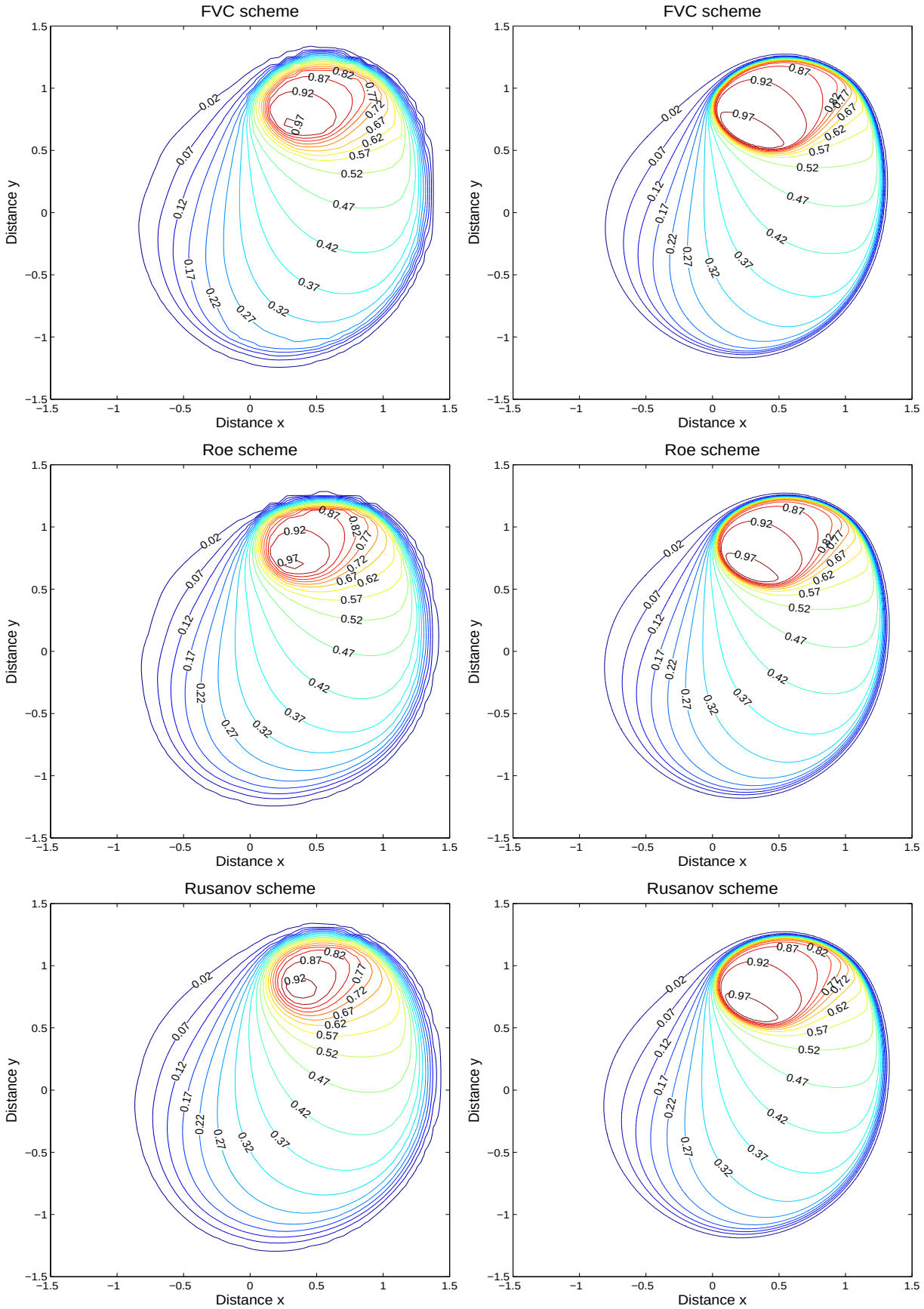

Figure 15: Results for the Buckley-Leverett problem at $t = 0.5$ using FVC scheme (first row), Roe scheme (second row) and Rusanov scheme (third row) using 50×50 gridpoints (left column) and 200×200 gridpoints (right column).

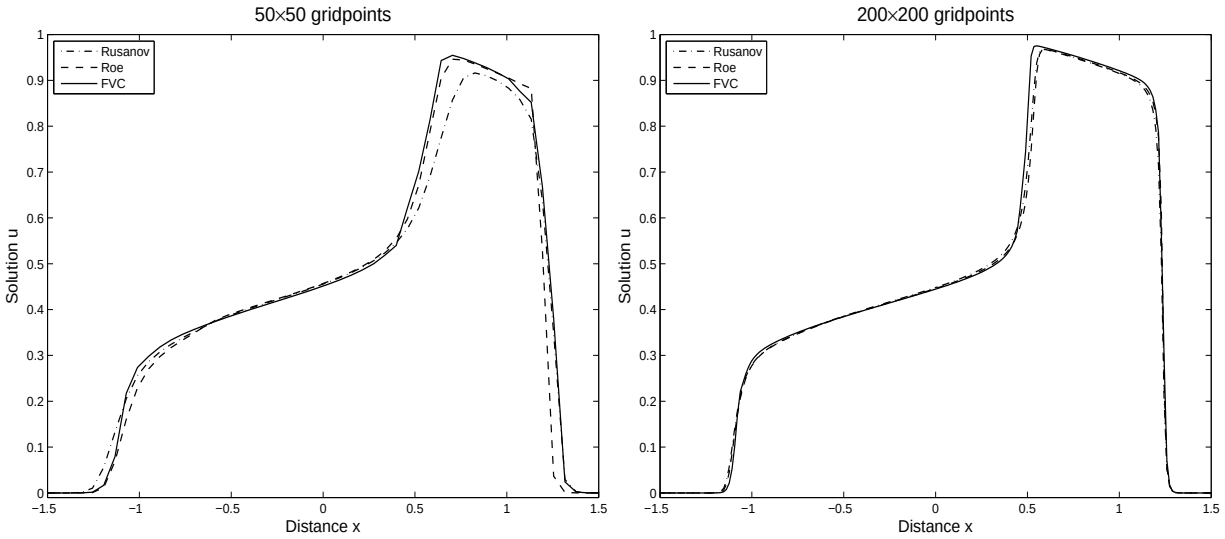

Figure 16: Cross sections at $y = 0.5$ of the results presented in Figure 15 for the Buckley-Leverett problem at $t = 0.5$ using 50×50 gridpoints (left column) and 200×200 gridpoints (right column).

5.4 Non-hyperbolic system

Our final concern is to ascertain the behaviour of the FVC scheme to solve non-hyperbolic systems of conservation laws. To this end we consider a 2×2 system of non-hyperbolic equations given as

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{w})}{\partial x} + \frac{\partial \mathbf{g}(\mathbf{w})}{\partial y} = \mathbf{0}, \qquad (x, y) \in [-2\pi, 2\pi] \times [-2\pi, 2\pi] \tag{31}$$

where

$$\mathbf{w} = \begin{pmatrix} u \\ \\ v \end{pmatrix}, \qquad \mathbf{f}(\mathbf{w}) = \begin{pmatrix} \frac{1}{2}u + \frac{1}{2}v \\ -\frac{1}{2}u + \frac{1}{2}v \end{pmatrix}, \qquad \mathbf{g}(\mathbf{w}) = \begin{pmatrix} \frac{1}{2}u + \frac{1}{2}v \\ -\frac{1}{2}u + \frac{1}{2}v \end{pmatrix}.$$

Initial and boundary conditions are derived from the analytical solution

$$\begin{aligned} u &= \sinh(t)\sin(x + y - t) + \sin(x - y), \\ v &= \cosh(t)\cos(x + y - t). \end{aligned}$$

The system (31) can be rearranged in an advective form as

$$\frac{\partial \mathbf{w}}{\partial t} + A\frac{\partial \mathbf{w}}{\partial x} + B\frac{\partial \mathbf{w}}{\partial y} = 0, \tag{32}$$

where A and B are the Jacobian matrices defined by

$$A = B = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{pmatrix}.$$

It is easy to verify that the two eigenvalues of the system are

$$\lambda_1 = \frac{1}{2} + \frac{1}{2}i, \qquad \lambda_2 = \frac{1}{2} - \frac{1}{2}i.$$

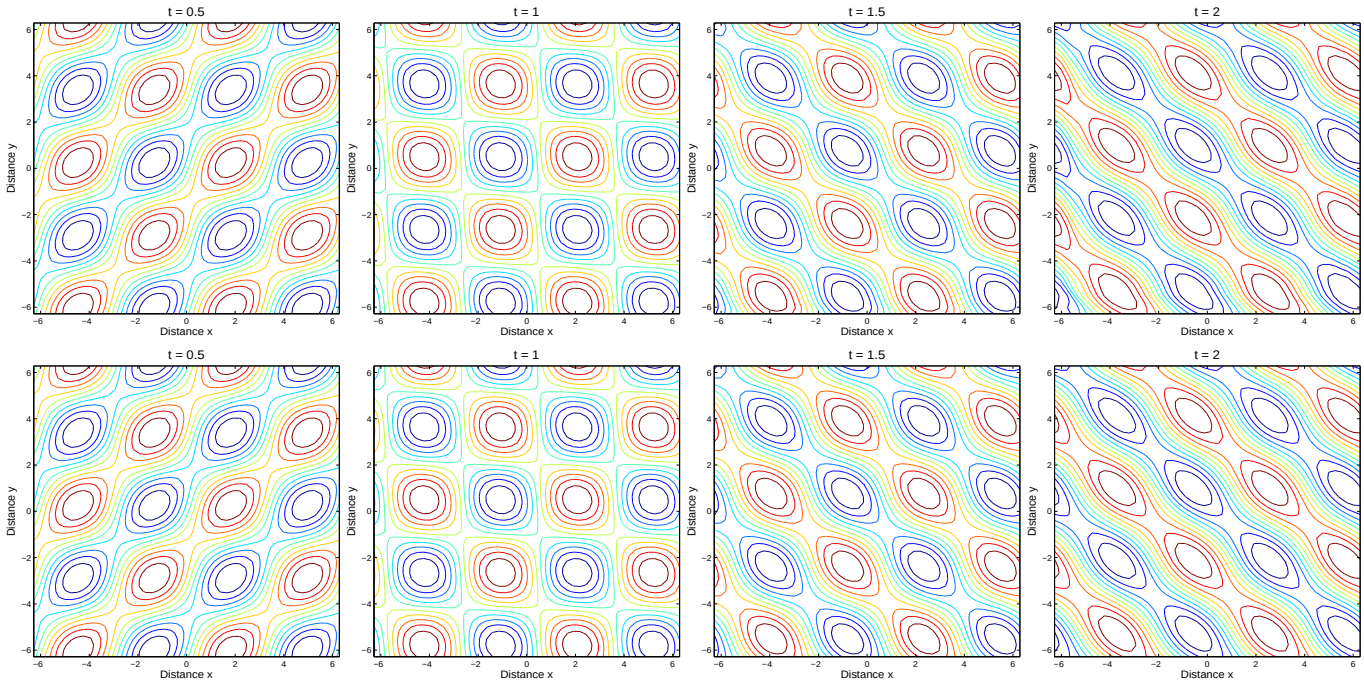

Figure 17: Results for the solution u in the non-hyperbolic system at four instants using FVC scheme (first row) and the exact solution (second row) using 50×50 gridpoints.

Table 3: L^{∞}-error in the non-hyperbolic system at $t = 0.5$ with two different interpolation procedures.

	Solution u			Solution v		
Interpolation	50×50	100×100	200×200	50×50	100×100	200×200
Bilinear	0.020984	0.010606	0.019663	0.058675	0.024471	0.022502
Bicubic	0.015965	0.008230	0.010559	0.037585	0.009115	0.007670

The aim of this example is to examine the performance of the proposed FVC scheme to solve non-hyperbolic systems of conservation laws. It should be emphasized that for this class of problems, the numerical methods based on exact or approximate Riemann solvers can not be applied. It is well-established that under certain physical conditions, many problems in geophysics such as two-phase flows and two-layer shallow water equations may exhibit a loss of hyperbolicity. Physically, the loss of hyperbolicity could be linked to the emergence of shear instabilities. From a mathematical point of view, it results in ill-posedness of the problem to be solved whereas numerically, traditional schemes that require eigenstructure and symmetrical treatment of the system under study will generally lead to instabilities. Note that for this test example the considered conservation laws are solved with a zero diffusion term. As a consequence, this problem is more difficult to handle; the results shown here illustrate the robustness of the FVC method. Furthermore, the considered non-hyperbolic system is a problem unsteady in nature; therefore, good numerical accuracy is required in order to capture the different phenomena present in its evolving solution. The FVC scheme shows high accuracy and good stability for this transient problem.

In Figure 17 we display the results obtained for the solution u at four different instants using a coarse mesh with 50×50 gridpoints. This figure also shows the analytical solution at the corresponding times $t = 0.5$, 1, 1.5 and 2. Apparently, the overall solution features for this example are preserved with no spurious oscillations appearing in the results obtained using the proposed FVC method. Obviously, the

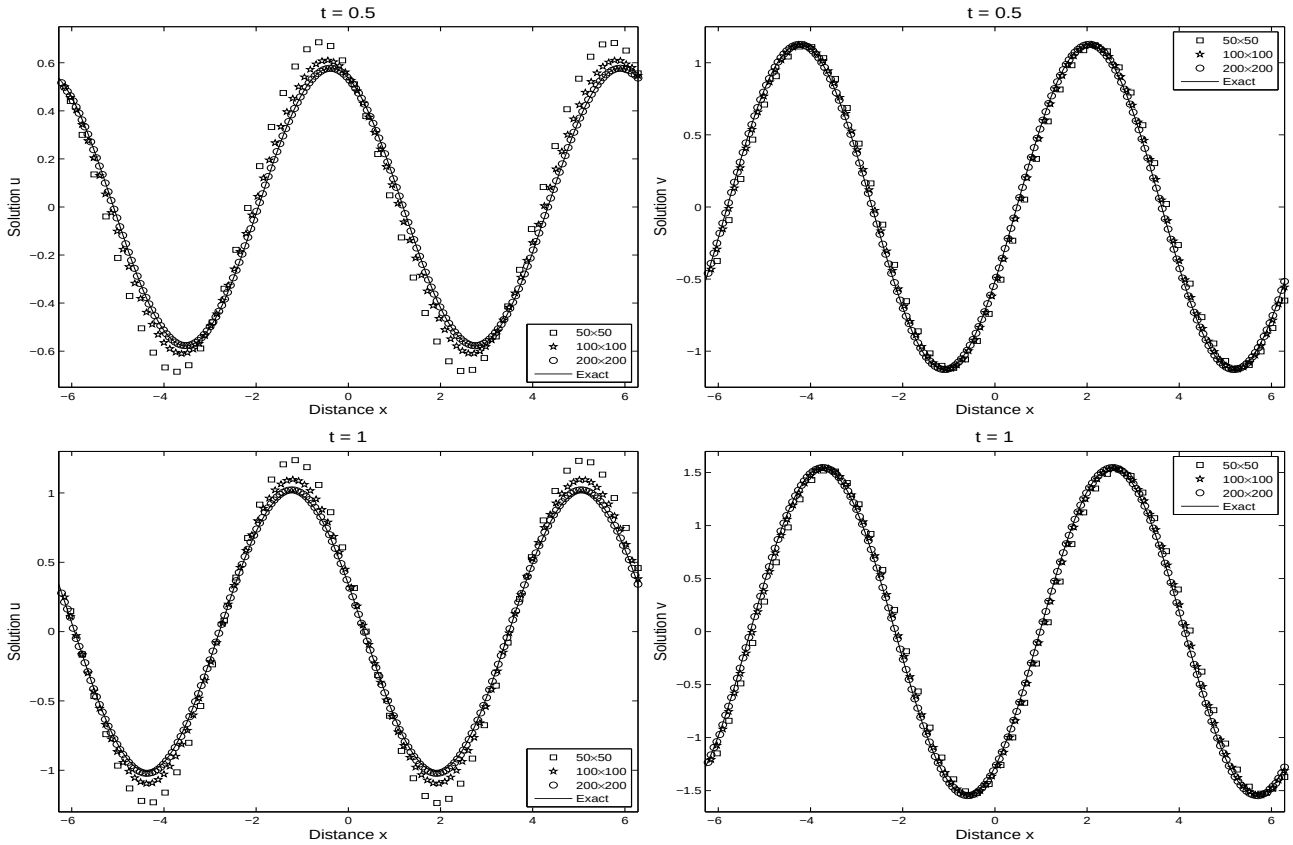

Figure 18: Cross sections at $y = -\frac{\pi}{2}$ of the results for the solution u (left) and the solution v (right) in the non-hyperbolic system at two instants $t = 0.5$ (top) and $t = 1$ (bottom) using different meshes.

computed results verify the stability and the shock capturing properties of the proposed FVC method. The periodic character of the component u in the figure should be noted. Similar features, not included here, have been detected in the solution component v. The FVC results are very satisfactory and show good agreement with the analytical solution. To illustrate the grid effects on the solution behaviour we present in Figure 18 cross sections at $y = -\frac{\pi}{2}$ of the solutions u and v at two instants $t = 0.5$ and $t = 1$ using different meshes. It is evident that an increase in the number of gridpoints results in a decrease of the numerical errors in both solutions u and v. A faster error decay has been observed in the results obtained for the solution v than those obtained for the solution u. The grid convergence is clearly achieved in the proposed FVC scheme that, using for example a fine mesh of 200×200 gridpoints, the FVC method was able to produce stable solutions independently of the grid effects.

As a final remark we check the response of the FVC scheme for the interpolation procedure used in the predictor stage. To this end, we summarize in Table 3 the L^∞-error in the solutions u and v at $t = 0.5$ using two different interpolation procedures namely, the bilinear and bicubic methods. Under the considered conditions for this non-hyperbolic system, the bicubic interpolation illustrates more accurate results than the bilinear interpolation. We have also observed that for fine meshes the error differences between the results obtained using the bilinear and bicubic interpolation procedures are negligible.

6 Conclusions

A simple and accurate family of finite volume Eulerian-Lagrangian methods has been developed for solving two-dimensional equations of nonlinear conservation laws. The proposed finite volume method combines the advantages of the finite volume discretization such as conservation property and of the modified

method of characteristics such as elimination of Riemann solvers. The method can also be interpreted as a predictor-corrector procedure to convert the conventional semi-Lagrangian methods to conservative and non-oscillatory. In the first stage, the scheme reconstructs the numerical fluxes using the modified method of characteristics in a Lagrangian framework. This stage results in an upwind discretization of the characteristic variables and avoids the Riemann problem solvers. In the second stage, the solution is updated using the Eulerian finite volume discretization of the conservation laws. The developed method does not require either linear or nonlinear solutions or special front tracking techniques. The performance of the method has been assessed for several test examples using pure advection, advection-diffusion, inviscid and viscous Burgers, and Buckley-Leverett problems. We have also compared the numerical results obtained using our method to those computed using conventional Roe and semi-Lagrangian methods. In the considered test problems, the proposed method has exhibited accurate solutions with correct conservation property and non-oscillatory behaviour. The presented results make it promising to be applicable also to real situations where, beyond the many sources of complexity, there is a more severe demand for accuracy in solving hyperbolic systems of conservation laws, which must be performed for long simulation times.

The proposed method has mainly been applied to scalar conservation laws. Clearly this is not the case for practical applications. Therefore, as future work, one must first extend the implementation of the method to hyperbolic and non-hyperbolic systems of conservation laws and then develop new methods capable of modeling realistic applications in multi-phase flows and multi-layer shallow water equations among others. Theoretically, the extend of the method to hyperbolic systems of conservation laws is straightforward. However, the number of the departure points will increase with increasing variables in the considered system. As a result, the future research should be focused on the development of new methods that can accurately calculate the characteristics with relatively low computational cost. The computational efficiency of the method can be further improved by advanced interpolation techniques and optimization of the code. A desirable study would also be a more thorough evaluation of the method accuracy in finite volume discretization on unstructured grids. The computational domains in these problems are more typical of realistic cases and is expected to be more interesting and serve as a better test of efficiency and accuracy.

Acknowledgment. Part of this work was carried out during a visiting stay of the second author in the School of Engineering and Computing Sciences at University of Durham.

Appendix: Conventional Eulerian and semi-Lagrangian methods

In this appendix we briefly describe the methods used in our study to assess the FVC results.

The Rusanov method [17]: Applied to the equation (1) the Rusanov scheme results in the conservative form (3) with the numerical fluxes are given by

$$\begin{aligned} F_{i+1/2,j} &= \frac{1}{2}\Big(f(U_{i,j}) + f(U_{i+1,j})\Big) - \frac{1}{2}A_{i+1/2,j}\Big(U_{i+1,j} - U_{i,j}\Big), \\ G_{i,j+1/2} &= \frac{1}{2}\Big(g(U_{i,j}) + g(U_{i,j+1})\Big) - \frac{1}{2}B_{i,j+1/2}\Big(U_{i,j+1} - U_{i,j}\Big), \end{aligned} \tag{33}$$

where $A_{i+1/2,j}$ and $B_{i,j+1/2}$ are the Rusanov velocities defined as

$$A_{i+1/2,j} = \max\Big(\big|f'(U_{i,j})\big|, \big|f'(U_{i+1,j})\big|\Big), \qquad B_{i,j+1/2} = \max\Big(\big|g'(U_{i,j})\big|, \big|g'(U_{i,j+1})\big|\Big).$$

The Roe method [15]: Applied to the conservation law (1) the Roe scheme is formulated in the conservative form as (3) with the numerical fluxes are

$$\begin{aligned} F_{i+1/2,j} &= \frac{1}{2}\Big(f(U_{i,j}) + f(U_{i+1,j})\Big) - \frac{1}{2}\left|\frac{f'(U_{i,j}) + f'(U_{i+1,j})}{2}\right|(U_{i+1,j} - U_{i,j}), \\ G_{i,j+1/2} &= \frac{1}{2}\Big(g(U_{i,j}) + g(U_{i,j+1})\Big) - \frac{1}{2}\left|\frac{g'(U_{i,j}) + g'(U_{i,j+1})}{2}\right|(U_{i,j+1} - U_{i,j}). \end{aligned} \tag{34}$$

The semi-Lagrangian method: In contrast to the proposed FVC method for which predictor and corrector stages are needed, the conventional semi-Lagrangian (SLAG) method consists of applying the modified method of characteristics directly to the advection equation (4). Following for example [5, 14], the characteristics of the equation (4) are the solutions of initial-value problem for ordinary differential equations

$$\begin{aligned} \frac{dX_i(\tau)}{d\tau} &= V\Big(\tau, X_i(\tau), Y_j(\tau)\Big), \qquad \tau \in [t_n, t_n + \Delta t], \\ \frac{dY_j(\tau)}{d\tau} &= W\Big(\tau, X_i(\tau), Y_j(\tau)\Big), \qquad \tau \in [t_n, t_n + \Delta t], \\ X_i(t_n + \Delta t) &= x_i, \qquad Y_j(t_n + \Delta t) = y_j, \end{aligned} \tag{35}$$

where $V = f'(U)$ and $W = g'(U)$. Here, $\mathbf{X}_{i,j}(\tau) = \big(X_i(\tau), Y_j(\tau)\big)^T$ is the departure point at time τ of a particle that will arrive at $\mathbf{x}_{i,j} = (x_i, y_j)^T$ at time t_{n+1}. Note that for the SLAG method, all the information on the numerical solution is allocated at the nodes of the cell. The implementation of the SLAG algorithm to solve (4) can be carried out in the following steps:

Step 1. *For all gridpoints* $\mathbf{x}_{i,j}$ *compute the departure points* $\mathbf{X}_{i,j}(t_n) = \big(X_i(t_n), Y_j(t_n)\big)^T$ *using an iterative procedure as in (11).*

Step 2. *Identify the element* $\mathcal{C}_{i,j}$ *of the numerical mesh where* $\mathbf{X}_{i,j}$ *belongs.*

Step 3. *Update the solution* $U_{i,j}^{n+1}$ *employing for example a Lagrange interpolation.*

It should be pointed out that, the SLAG scheme is stable independently of the spatial stepsizes Δx and Δy, so that the choice of Δt is based only on the condition (12). In addition, it is well-known that the SLAG scheme is not monotone and it fails to conserve mass, see for instance [19] and further references are therein.

References

[1] F. Benkhaldoun and M. Seaid, Combined characteristics and finite volume methods for sediment transport and bed morphology in surface water flows, *Math. Comput. Simulat.*, **81**, 2073-2086 (2011).

[2] F. Benkhaldoun and M. Seaid, A simple finite volume method for the shallow water equations, *J. Comput. Appl. Math.*, **234**, 58–72 (2010).

[3] R. Bermejo, *A Galerkin-characteristic algorithm for transport-diffusion equations*, SIAM J. Numer. Anal. **32** (1995) pp. 425–454.

[4] I. Christov and B. Popov, *New nonoscillatory central schemes on unstructured triangulations for hyperbolic systems of conservation laws*, J. Comput. Phys. **227** (2008) pp. 5736–5757.

[5] J. Douglas, and T.F. Russell, *Numerical methods for convection dominated diffusion problems based on combining the method of characteristics with finite elements or finite differences*, SIAM J. Numer. Anal. **19** (1982) pp. 871–885.

[6] R.E. Ewing, H. Wang, A summary of numerical methods for time-dependent advection-dominated partial differential equations, *J. Comput. Appl. Math.*, **128**, 423–445 (2001).

[7] S.V. Krisnamachari, L.J. Hayes, and T.F. Russel, *A finite element alternating-direction method combined with a modified method of characteristics for convection-diffusion problems*, SIAM J. Numer. Anal. **26** (1989) pp. 1462–1473.

[8] R.J. LeVeque, Numerical methods for conservation laws, Lectures in Mathematics ETH Bürich, (1992).

[9] J. Liu, H. Chen, R.E. Ewing, and G. Qin, *An efficient algorithm for characteristic tracking on two-dimensional triangular meshes*, Computing. **80** (2007) pp. 121–136.

[10] K.W. Morton, *Numerical Solution of Convection-Diffusion Problems*, Chapman & Hall, London, 1996.

[11] T.N. Phillips, A.J. Williams, A semi-Lagrangian finite volume method for Newtonian contraction flows. SIAM J. Sci. Comput. **22**, 2152-2177 (2001)

[12] J. Pudykiewicz and A. Staniforth, Some properties and comparative performance of the semi-Lagrangian method of Robert in the solution of advection-diffusion equation, *Atmos. Ocean.*, **22**, 283–308 (1984).

[13] P.A. Raviart, E. Godlewski, *Hyperbolic Systems of conservation laws*, collection Mathematiques et Applications, SMAI, Ellipses Eds, N 3/4, 1990.

[14] A. Robert, *A stable Numerical Integration Scheme for the Primitive Meteorological Equations*, Atmos. Ocean **19** (1981) pp. 35–46.

[15] P.L. Roe, Approximate Riemann solvers, parameter vectors and difference schemes, J. Comput. Phys. **43**, 357-372 (1981).

[16] H. Rui, A conservative characteristic finite volume element method for solution of the advection-diffusion equation. Comput. Methods in Appl. Mech. Eng. **197**, 3862-3869 (2008)

[17] V. Rusanov, Calculation of interaction of non-steady shock waves with obstacles. J. Comp. Math. Phys. USSR. **1**, 267–279 (1961)

[18] T.F. Russell and M.A. Celia, An overview of research on Eulerian-Lagrangian localized adjoint methods (ELLAM). Adv. Water Resour. **25**, 1215–1231 (2002)

[19] M. Seaid, On the quasi-monotone modified method of characteristics for transport-diffusion problems with reactive sources, *Comp. Methods in Appl. Math.*, **2**, 186–210 (2002).

[20] E. Süli, *Convergence and stability of the Lagrange-Galerkin method for the Navier-Stokes equations*, Numer. Math., **53** (1988), pp. 459–483.

[21] C. Temperton and A. Staniforth, An efficient two-time-level semi-Lagrangian semi-implicit integration scheme, *Quart. J. Roy. Meteor. Soc.*, **113**, 1025–1039 (1987).